



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA STROJNÍHO INŽENÝRSTVÍ**

FACULTY OF MECHANICAL ENGINEERING

**ÚSTAV AUTOMATIZACE A INFORMATIKY**

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

**IMPLEMENTACE SYSTÉMU PRO  
ANALÝZU A OPTIMALIZACI OBCHODNÍCH  
SYSTÉMŮ S VYUŽITÍM UMĚLÉ  
INTELIGENCE**

IMPLEMENTATION OF THE SYSTEM FOR ANALYSIS AND OPTIMIZATION OF  
BUSINESS SYSTEMS WITH THE USE OF ARTIFICIAL INTELLIGENCE

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

Bc. Kamil Staněk

**VEDOUCÍ PRÁCE**

SUPERVISOR

prof. RNDr. Ing. Jiří Šťastný, CSc.

BRNO 2019



# Zadání diplomové práce

Ústav: Ústav automatizace a informatiky  
Student: **Kamil Staněk**  
Studijní program: Strojní inženýrství  
Studijní obor: Aplikovaná informatika a řízení  
Vedoucí práce: **prof. RNDr. Ing. Jiří Šťastný, CSc.**  
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

## **Implementace systému pro analýzu a optimalizaci obchodních systémů s využitím umělé inteligence**

### **Stručná charakteristika problematiky úkolu:**

Cílem diplomové práce bude realizace obchodní zakázky zaměřené na vytvoření systému pro analýzu a optimalizaci burzovních automatických obchodních systémů (AOS) s využitím metod umělé inteligence. Dílčími cíli bude otestování konkrétních AOS a jejich podrobná analýza.

### **Cíle diplomové práce:**

Vytvoření systému rychlé analýzy vycházející z reálných obchodních dat měnových párů.  
Předpokládaná realizace v programu MATLAB.

Úprava zákaznických AOS do prostředí, které umožní analýzu ve výše uvedeném systému.

Vytvoření prediktivního systému s využitím metod umělé inteligence pro vývoj cen na základě historických obchodních dat konkrétních měnových párů.

Doplnění systému o grafické výstupy.

Výstupní podrobná analýza na základě simulace reálného obchodování s historickými daty.

### **Seznam doporučené literatury:**

LIEN, K. Forex: ziskové intradenní a swingové obchodní strategie : jak na technickou a fundamentální analýzu pro úspěch na finančních trzích. 2., rozš. vyd. Praha: FXstreet, 2013. ISBN 978-80-904418-2-8.

OLSON, David L. a Desheng WU. Predictive Data Mining Models (Computational Risk Management). 1. Switzerland: Springer, 2017. ISBN 978-9811025426.

REJNUŠ, O. Finanční trhy. 4., aktualiz. a rozš. vyd. Praha: Grada, 2014. Partners. ISBN 978-80-247-3671-6.

SMITH, J. PREDICTIVE ANALYTICS with NEURAL NETWORKS using MATLAB. 1. USA: CreateSpace Independent Publishing Platform, 2017. ISBN 978-1544169613.

WALKER, W. Expert Advisor Programming and Advanced Forex Strategies. 1. Michigan: Independently published, 2019. ISBN 978-1726669788.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19.

V Brně, dne 26. 10. 2018



\_\_\_\_\_  
doc. Ing. Radomil Matoušek, Ph.D.  
ředitel ústavu

\_\_\_\_\_  
V. Z. Katolický  
doc. Ing. Jaroslav Katolický, Ph.D.  
děkan fakulty

## **ABSTRAKT**

Diplomová práce se zabývá realizací obchodní zakázky spočívající ve vytvoření systému rychlé analýzy a optimalizace vycházející z reálných obchodních dat měnových párů v programu MATLAB. V rámci zpracovávaného tématu je implementován zákaznický automatický obchodní systém včetně analýzy a optimalizace parametrů systému. Byl také využit prediktivní systém vývoje cen s použitím metod umělé inteligence na základě historických dat. Získaná data z optimalizací a simulací automatických obchodních systémů jsou přehledně shrnuty do grafických výstupů.

## **ABSTRACT**

This diploma thesis deals with the realization of a business order, consisting in creating a system of rapid analysis and optimization based on real business data of currency pairs using the MATLAB software. Within this developed topic a customer automated trading system is implemented, including the analysis and the optimization of the system parameters. There was also used a predictive system of the development of the prices, using artificial intelligence methods, based on historical data. The obtained data from the optimization and the simulations of the automated trading system are clearly summarized in graphic outputs.

## **KLÍČOVÁ SLOVA**

Automatický obchodní systém, MATLAB, LSTM, simulace obchodování, měnové páry, prediktivní systém.

## **KEYWORDS**

Automated trading system (ATS), MATLAB, LSTM, trading simulation, currency pairs and a predictive system.



## BIBLIOGRAFICKÁ CITACE

STANĚK, Kamil. *Implementace systému pro analýzu a optimalizaci obchodních systémů s využitím umělé inteligence*. Brno, 2019. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/117301>. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automatizace a informatiky. Vedoucí práce Jiří Šťastný.





## **PODĚKOVÁNÍ**

Děkuji vedoucímu diplomové práce prof. RNDr. Ing. Jiřímu Šťastnému, CSc. za odbornou pomoc a velmi cenné rady při zpracování této diplomové práce.



## **ČESTNÉ PROHLÁŠENÍ**

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením prof. RNDr. Ing. Jiřího Šťastného, CSc. a s použitím literatury uvedené v seznamu literatury.

V Brně dne 23. 5. 2019

.....

Kamil Staněk



# OBSAH

<b>1</b>	<b>ÚVOD.....</b>	<b>15</b>
<b>2</b>	<b>PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ.....</b>	<b>17</b>
2.1	Obchodování na burze .....	17
2.1.1	FOREX .....	18
2.1.2	Zaznamenávání cen v grafu .....	18
2.2	Automatický obchodní systém .....	19
2.3	Testery .....	20
2.4	Prediktivní systém .....	20
2.4.1	Rekurentní neuronové sítě (RNN) .....	21
2.4.2	LSTM neuronové sítě .....	21
2.5	Software MATLAB .....	22
<b>3</b>	<b>VLASTNÍ ŘEŠENÍ.....</b>	<b>25</b>
3.1	Požadavky zákazníka.....	25
3.2	Strategie postupu vývoje automatického obchodního systému .....	26
3.2.1	Zachycení myšlenky AOS .....	26
3.2.2	Hrubý vývoj .....	27
3.2.3	Upřesnění parametrů.....	27
3.2.4	Optimalizace .....	27
3.2.5	Analýza dat .....	28
3.3	Vývoj testovací části systému.....	28
3.3.1	Zajišťování TICK dat .....	31
3.3.2	Ukládání dat.....	32
3.3.3	Vytvoření modulu pro výběr testovacích dat .....	34
3.3.4	Vytvoření modulu pro zobrazování pomocí svíček.....	35
3.3.5	Zpracování signálu pro požadovanou strategii .....	37
3.4	Vývoj simulační části systému .....	40
3.5	Optimalizace .....	42
3.6	Analýza výsledků .....	46
3.7	Predikce dat .....	49
<b>4</b>	<b>ZHODNOCENÍ A DISKUZE.....</b>	<b>55</b>
<b>5</b>	<b>ZÁVĚR .....</b>	<b>57</b>
<b>6</b>	<b>SEZNAM POUŽITÉ LITERATURY.....</b>	<b>59</b>
<b>7</b>	<b>SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK .....</b>	<b>61</b>



# 1 ÚVOD

Obchodování na burze je velkým lákadlem pro mnoho lidí. Burza je místem, kde se střetává poptávka s nabídkou různých institucí i samotných traderů. Účastníci burzovních obchodů se na burze pohybují s úmyslem prodat nebo nakoupit konkrétní komodity, akcie nebo měny. Typů burz je mnoho, každá z nich má svá specifická pravidla a většina z nich se specializuje jen na konkrétní typy obchodů.

Vzhledem k tomu, že se obchodování na burze věnuji již 11 let, rozhodl jsem se tyto své znalosti a zkušenosti zúročit při realizaci konkrétní obchodní zakázky. Celý proces realizace této zakázky jsem zúročil i při zpracování tohoto tématu diplomové práce. Požadavky zákazníka spočívaly ve vytvoření systému pro analýzu a optimalizaci burzovních automatických obchodních systémů (AOS).

Dalším důvodem pro realizaci této zakázky byl i fakt, že mnoho traderů neustále vidí různé reklamy a nabídky, které ho naučí ziskově obchodovat. Už z principu je nelogické, aby tomu tak bylo. Kdyby sám předkladatel tohoto řešení uměl neustále obchodovat vysoce ziskově, nedělal by školení pro ostatní. Pořádaná školení zaměřená na tuto problematiku mají diametrálně odlišnou úroveň. Vždy byl předložen zaručený recept na trvalé generování zisku pomocí jejich AOS, na základě kterého sami přednášející ziskově obchodují již několik let. V okamžiku, kdy účastníci školení kladou podrobné dotazy na nastavení jejich AOS, je zjištěno, že je využívána i intuice, která do systému nemůže být zahrnuta.

Tato skutečnost se zřetelně projevila na absolvovaných on-line obchodováních, které trvaly dvakrát pět dnů. Zde se v reálu ukázalo, že obchodování je především o zkušenostech, emocích a dodržování disciplíny, neboť většina systémů nešla převést na mechanické obchodování. Tato osobní zkušenost přispěla k myšlence realizovat testovací software, který dokáže rychle analyzovat vložené strategie a vyhodnotit je, zdali má smysl se jimi dále zabývat. Čas investovaný do testování je vysoce důležitý, neboť trader nemá následně potřebu zkoušet obchodování ručně na live obchodním účtu, aniž by měl možnost ověřit své hypotézy.

Pro naplnění tohoto cíle bude využit program MATLAB od firmy MathWorks. Myšlenku automatického obchodního systému, která bude v tomto programu realizována, byla kompletně dodaná zákazníkem. Tento AOS byl upraven tak, aby umožnil analýzu a optimalizaci nově navrženého systému. Na podnět zákazníka byla využita i predikce cen s využitím metod umělé inteligence na datech konkrétního měnového páru. Součástí požadavků zákazníka jsou i grafické výstupy zachycující chování automatického obchodního systému na různé vstupní parametry při jeho simulaci obchodování. K tomuto účelu jsou využita reálná historická data konkrétních měnových párů získaných od firmy MetaQuotes.

Práce je rozdělena na dva hlavní celky. V první teoretické části bude shrnut aktuální stav poznání v oblasti obchodování na burze a testování automatických obchodních systémů. Taktéž bude popsán software MATLAB pro ucelenost informací

potřebných k pochopení principů mnou vytvořeného testovacího systému. Pozornost dále bude zaměřena na popis procesu vytvoření prediktivního systému s využitím umělé inteligence.

Druhá, praktická část této práce již bude zaměřena na konkrétní popis vývoje tohoto testovacího systému včetně provedených analýz a optimalizací chodu celého systému.

V závěrečném zhodnocení bude popsána využitelnost vytvořeného systému v reálném prostředí včetně kritického zhodnocení očekávaných výsledků.



## 2 PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ

V rámci současného stavu poznání z obchodování na burze budou následně v jednotlivých podkapitolách popsány základní informace z oblasti využití automatických obchodních systémů a jejich testování. Poté bude pozornost zaměřena na predikování dat pomocí umělé inteligence. Taktéž bude popsána využitelnost softwaru MATLAB pro analýzu a predikci dat.

### 2.1 Obchodování na burze

Burzy lze rozdělit podle mnoha kritérií, například podle investičního nástroje na:

- Akciové
- měnové
- derivátové
- plodinové a jiné. [1]

Dle dalšího členění lze burzy rozdělit podle významu a typu obchodování. V této práci se dále zaměřím pouze na celosvětové burzy a výhradně s elektronickým typem obchodování. Důvodem je fakt, že tento typ obchodování na burzách provádí zákazník, který si objednal vytvoření testovacího systému. Následně tedy bude veškerá práce s těmito daty prováděna pouze na měnové burze FOREX, ačkoliv použití nově vytvořeného testovacího systému umožňuje jeho použití i pro jiné investiční nástroje.

Existuje mnoho obchodních platform, pomocí kterých lze obchodovat a ve většině případů má každý broker svou upravenou platformu. Mezi nejznámější patří například MT4, MT5, NinjaTrader, MetaStock, JForex (Dukascopy), e-Broker (FIO).

Pokud potřebujeme makroekonomická data pro fundamentální analýzu, je velmi vhodné použít nějaký makroekonomický kalendář. Velmi využívaný je kalendář od firmy Fair Economy, Inc., zobrazitelný na stránkách [www.forexfactory.com](http://www.forexfactory.com). [12] Výhoda využívání makroekonomických kalendářů se projeví zejména při ručním obchodování, kdy lze přestat obchodovat v době zásadních makroekonomických dat, například změně úrokových sazeb a podobně.

Na burze lze na těchto platformách obchodovat pomocí dvou způsobů, a to diskrečním obchodováním, které je plně v režii obchodníka a je zadáváno ručně. Druhý způsob obchodování je mechanický, což znamená, že roli obchodníka zastupuje software. Zásadním rozdílem mezi uvedenými způsoby je, že při diskrečním obchodování má obchodník aktivní kontrolu své obchodní strategie na základě jeho intuice. Při mechanickém obchodování je užití intuice vyloučeno. V následujících kapitolách se již budu věnovat pouze mechanickému obchodování.

### 2.1.1 FOREX

Volba zákazníka se zaměřením na tento trh je zcela logická, neboť se jedná o největší a nejlikvidnější trh na světě. Odhaduje se, že každý den se zde zobchodují biliony dolarů. Jde o decentralizovaný trh předurčený především pro síť bank a Market Makerů. Cílem obchodníků je využití mnoha mechanismů na zhodnocení investic. Rovněž se tento trh používá k diverzifikaci rizika, tzv. hedge obchody. Tyto obchody nevyužívají malí obchodníci, neboť ti se zabývají vyděláváním na pohybech konkrétních měn, respektive kurzů mezi měnami. [7]

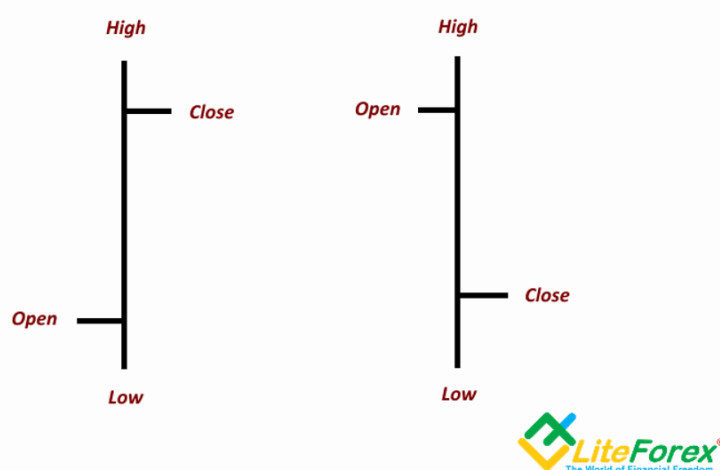
Na tomto trhu jde o spekulaci a obchodování měn. Devizový trh je otevřen 24 hodin denně, 5 dní v týdnu. Existuje základní měna versus kótovací měna. Kombinace těchto dvou měn vytváří měnový pár. Například EURUSD, USDJPY, GBPUSD a jiné. [1]

Významným přínosem pro využití tohoto trhu pro provozovatele automatických obchodních systémů jsou velmi nízké náklady a otevření obchodní pozice. Prakticky všichni brokeri umožňují obchodníkovi použití demo účtu pro vyzkoušení své obchodní platformy s možností vyzkoušet si obchody nanečisto.

### 2.1.2 Zaznamenávání cen v grafu

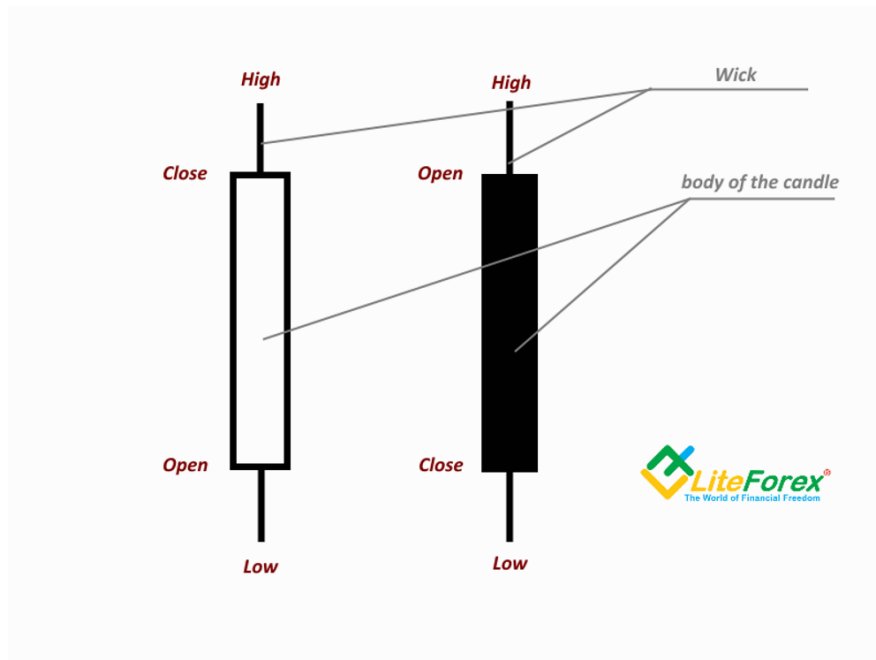
Existuje mnoho způsobů vyjádření cen v grafu. Nejčastější z nich je cena určená pomocí svíček. Následně uvádím dva nejvyužívanější typy zobrazování, které lze navíc zobrazit i v programu MATLAB.

Japanese Candlestick charts znázorňuje jednoduché a přehledné zobrazení všech parametrů pro konkrétní časový úsek označovaného jako Time Frame (TF). [9]



Obrázek 1 Japanese Candlestick chart [9]

Heikin Ashi je jiné vyjádření výše uvedených svíček. Jedná se o nejvyhledávanější zobrazení cen na grafech mezi tradery, neboť změnou barvy vnitřního těla svíčky je ihned určen směr pohybu ceny. Bílá je nejčastěji LONG (cena roste) a černá nebo tmavá barva značí SHORT (cena klesá). [9]



Obrázek 2 Heikin Ashi [9]

## 2.2 Automatický obchodní systém

Za automatický obchodní systém (AOS) můžeme považovat jakoukoli strategii, která je zcela nezávislá na uživatelském rozhodnutí, pokud je spuštěna. Parametry pro tento AOS se zadávají vždy dopředu, ale mohou se v průběhu chodu strategie měnit podle předem známých pravidel. Tento systém si sám tyto vstupy upravuje a jejich vyhodnocení je výstupem jeho činnosti, což v konečném důsledku znamená, že obchoduje za uživatele. To, jakým způsobem systém vyhodnocuje vstupy (obvykle historická data), určuje vlastní obchodní strategii tohoto systému. Samotný AOS nemusí vyhodnocovat pouze historická data, může použít mnoho indikátorů, z nichž jsou patrně nejznámější klouzavé průměry (MA) a jejich křížení. Použití indikátorů nebo historických dat je zcela na rozhodnutí tvůrce AOS. K největším výhodám patří absence jakýchkoli emocí při obchodování, kdy AOS provádí obchody podle předem přesně stanovených pravidel. K dalším výhodám patří rychlost systému, která umožní i obchodování velmi malých rozdílů pohybů obchodních cen, tzv. scalping. [8]

## 2.3 Testery

Tester je v tomto kontextu uvažován jako software, který na základě daných pravidel provádí činnost mechanického obchodování na historických datech. Tato činnost je plně automatizována a slouží k tomu, aby bylo možné zjistit, jakých výsledků dosahuje nastavení jedné konkrétní automatické obchodní strategie. Změnou parametrů a spuštěním AOS s těmito novými parametry dojde k novému obchodování a analýze těchto jiných parametrů. Pomocí kombinace určité sekvence předem daných vstupních změn na parametrech a spuštění testování AOS s postupnou změnou těchto parametrů, dojde k získání výsledků pro každou konkrétní variantu vstupních dat. Tato vstupní data jsou následně analyzována a dochází k optimalizaci nastavení automatických obchodních systémů.

Tato činnost je základem pro všechny testery, pomocí kterých získá tvůrce AOS zpětnou vazbu a umožňuje mu modulaci parametrů AOS. Pokud nedochází k ziskovosti z konkrétní sady dat, i to je důležitá informace pro obchodníka, neboť musí dojít k významným změnám v rozhodovacím procesu zadávání obchodu AOS.

Za nejznámější tester mezi malými obchodníky lze zcela jistě považovat tester zabudovaný přímo v obchodní platformě MT4 od firmy MetaQuotes. MT4 nejen že umožňuje samotné obchodování, jak je od něj očekáváno, ale právě ona funkcionality testeru z něj dělá ideální platformu pro základní seznámení se s AOS. Nevýhodou tohoto testeru je ovšem jeho velmi dlouhý čas testování a nutnost používat speciální programovací jazyk MQL4 (MetaQuotes Language). Následníkem MT4 je MT5 od stejné firmy, který ale zdaleka nedosahuje takové popularity, jako jeho předchůdce z důvodu neschopnosti zadávat multiplikované obchody jedním směrem.

Jako příklad nesprávně zvládnuté optimalizace chodu testeru může posloužit i příklad z mé praxe, kdy jeden ze systémů bylo nutno testovat na šesti počítačích po dobu 14 dnů. I toto je jeden z důvodů, proč se zabývám vývojem vlastního testeru.

Existují i další testovací a simulační programy, ale jde téměř výhradně o komerční software, kde je vždy největší problém získání přesných historických dat, neboť není známo o jaká data a trh se jedná. Je možné použít i nekomerční softwary, které byly vyvinuty v C++ jazyku, jako například mnou dříve využívaný program FXSim od Jána Lučanského, který již není v současnosti možné instalovat.

## 2.4 Prediktivní systém

Pro znalost budoucího chování cenových pohybů je vyvíjena řada různých metod. Používají se na to heuristické analýzy, klasické algoritmy až po možnost strojového učení. Největší využití těchto metod přináší v ekonomické sféře predikce cen akcií, komodit a kurzů měn. Pro tvorbu prediktivního systému je nutné nejdříve zajistit návaznost časových řad a specifikovat daný predikční problém. Následně je třeba

správně zvolit predikční metodu, časový horizont a způsob úpravy dat pro výpočet. Z toho důvodu je vyžadována hluboká znalost burzovního trhu.

Za predikci můžeme označit proces začínající před událostí označovaný jako výpočet predikce. Další částí je poté probíhající proces tzv. predikce korekce. Proces je zakončen vyhodnocením predikce. Pro vyhodnocení kvality predikčního modelu je zásadní znalost hodnot skutečných a predikovaných. Pro toto vyhodnocení je mnoho kritérií, z nichž je nejčastěji využívána střední hodnota kvadratické chyby MSE – mean squared error, střední absolutní chyba MAE – mean absolute error a střední relativní chyba MAPE – mean absolute percentage error. Všechny tyto hodnoty jsou vyjádřeny v procentech – výpočet je proveden na základě následujících vzorců, kde je  $x_t$  naměřená hodnota a  $\hat{x}_t$  s tečkou je hodnota vypočtena predikcí. [10]

$$MSE = \frac{1}{N} \sum_{t=1}^N (x_t - \hat{x}_t)^2$$

$$MAE = \frac{1}{N} \sum_{t=1}^N |x_t - \hat{x}_t|$$

$$MAPE = \frac{100}{N} \sum_{t=1}^N \left| \frac{x_t - \hat{x}_t}{x_t} \right|$$

#### 2.4.1 Rekurentní neuronové sítě (RNN)

Rekurentní architektura neuronových sítí představuje velmi silný výpočetní model, který je schopen adaptovat se na prakticky libovolnou dynamiku. Je založena na vyhodnocování vztahů mezi aktuálními vstupy a zároveň mezi vstupy minulými. Tato síť bývá zpravidla vybavena cykly. Architektura RNN je velmi podobná struktuře lidského mozku a to mnohem více než klasická dopředná architektura. Z hlediska učení sítě však představuje mnohem obtížnější optimalizační problém než například síť dopředná, bez ohledu na to, zda jde o tradiční „Back-propagation through time“ nebo pomocí „Real-Time Recurrent Learning“. Časově zpětně putující chybové signály mají tendenci zmizet nebo explodovat. Pokud se jedná o klasickou verzi sítě s rekurentní architekturou a posloupnost je delší než 20 prvků, je prakticky nemožné dosáhnout rozpoznání vzorců ve vstupních datech. [15]

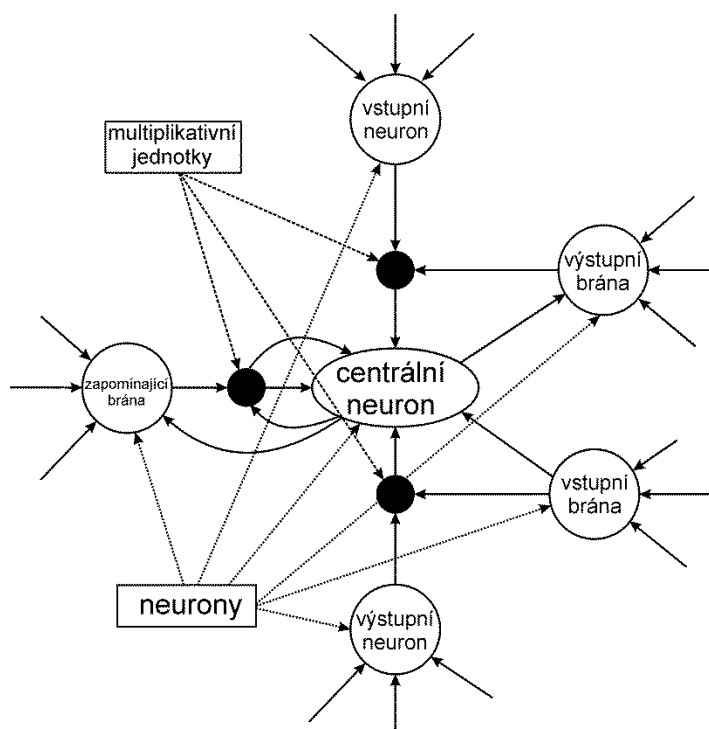
#### 2.4.2 LSTM neuronové sítě

Jde o síť s rekurentní architekturou, která realizuje krátkodobou i dlouhodobou paměť, jejíž základní jednotkou je paměťový blok nahrazující neurony ve skryté vrstvě tradiční RNN. Tím představují způsob, jak se vypořádat s mizejícím gradientem. Klasické rekurentní sítě jsou schopny rozpoznat vzorce ve vstupních datech pouze v malém časovém úseku kvůli mizejícímu nebo explodujícímu gradientu. LSTM síť je schopná

rozeznat vzorce chování při poskytnutí 1000 vzorků. Ve starší specifikaci šlo pouze o dvě řídicí jednotky. Současná specifikace je rozšířena o zapomínající mechanismus a síť tedy tvoří 3 řídicí jednotky nazývané brány. Brány jsou reprezentované neurony a řídí vstup, výstup a zapomínání centrálního neuronu, ve kterém může být hodnota uchována delší dobu. Každá z bran je spojena s multiplikativní jednotkou realizující součin všech svých vstupních hodnot. Brány ovlivňují hodnoty jejich vstupů, které slouží k interakci s ostatními prvky neuronové sítě. [15]

Funkce jednotlivých bran:

- Vstupní brána - rozhoduje, jestli se bude vstupní hodnota propagovat do centrální jednotky
- výstupní brána - rozhoduje, jestli se bude výstup z centrální jednotky propagovat dále
- zapomínající brána - rozhoduje, jestli bude hodnota uchována v centrální jednotce.



Obrázek 3 LSTM neuronová síť [vlastní zpracování]

## 2.5 Software MATLAB

Jde o zkratku slov matrix laboratory – MATLAB. Jedná se o programové prostředí s vlastním jazykem čtvrté generace. Jedná se o problémově orientovaný jazyk. Předností je rychlost výpočtů ve více vláknech a vynikající implementace práce s maticemi. K dalším přednostem patří vysoký počet možností grafických výstupů a analýz. Dalším využitím tohoto systému je přímé využití umělé inteligence, které je již

implementováno v programu MATLAB. Rovněž umožňuje využít grafické uživatelské rozhraní.

K nevýhodám rozhodně patří fakt, že se nejedná o komerční software, kdy jeho komerční využitelná varianta začíná na ceně 2 tisíce EUR ročně s updatem 800 EUR ročně. Vysoká cena jej předurčuje spíše k profesionálnímu využití. Výrobce si toho je vědom a nabízí i variantu pro užití v domácnosti za 119 EUR. Tato verze však nesmí být využita ve státní správě. Vyloučeno je i akademické či komerční využití v organizacích. Z tohoto důvodu je možné použít pro malé tradery i verzi Home, jakožto základ pro analýzu a testování, neboť samotný program žádný zisk nedokáže generovat. Proto také byl zvolen tento program pro tvorbu testeru dle požadavku zákazníka.

Základní verze je možné rozšiřovat pomocí Toolboxů v cenách od 35 EUR. V navrhovaném systému je použitý pouze Financial Toolbox, neboť na část využívající umělou inteligenci již není potřeba dokupovat další modul. Tento Toolbox je již součástí základní verze 2019. [11]

Při volbě MATLABu není třeba brát ohled na komerčnost, neboť není počítáno s přímým obchodováním ve vytvořeném systému ani ovládáním jakéhokoli jiného programu.

Vytvářený program dokáže udělat v této verzi pouze analýzu AOS a jeho nastavení. Do nastavení AOS mnou vytvořený program nijak nezasahuje. Nekontruluje tato nastavení, neboť to není jeho cílem. To umožňuje použití tohoto softwaru jak v domácnostech, tak pro zájmové rekreační obchodování, a to i na jeho demo účtech. Komerční efekt lze spatřovat maximálně v tom, že trader získá informaci o použitelnosti o jeho AOS v případě, že by jeho nepodloženou aplikací pro reálné obchodování nastala ztráta a on prodělal.

V případě, že trader může otestovat svůj systém dostatečně dopředu, nezíská sice nic, ale může dojít k efektu komerčního zisku, ačkoli tomu tak není, neboť se žádný zisk negeneruje.





### 3 VLASTNÍ ŘEŠENÍ

V této části textu bude popsáno vlastní řešení mojí práce. Nejdříve se zaměřím na požadavky zákazníka, které byly podrobně konkretizovány. Další nedílnou součástí je strategie postupu vývoje automatického obchodního systému. Dále se budu věnovat testovací části systému a na ni navážu popisem vývoje simulační části systému. Pozornost bude také zaměřena na optimalizaci, analýzu výsledků a predikci dat pomocí neuronové sítě.

#### 3.1 Požadavky zákazníka

V rámci mnou pořádaného kurzu výuky, kde jsem prezentoval techniky Price Action bez indikátorů i s klasickými indikátory, se na mě obrátil jeden z účastníků s žádostí o pomoc při výrobě automatického obchodního systému. Zákazník měl ucelenou představu o tom, jak by jeho systém mohl pracovat, avšak nedokázal naprogramovat tento AOS v jazyce MQL. Neměl tudíž možnost tento systém otestovat a zjistit jeho funkčnost na historických datech.

Při specifikaci požadavku zákazníka byl získán princip vyhodnocování vstupních dat AOS a jeho předpokládané nastavení. Pro otestování schopnosti vytvořit funkční a rychlý systém včetně optimalizačních technik, byly dodány zjednodušené parametry strategie, které umožní tento test provést. Jedná se o techniku, která je uváděna i pod názvem TLB (Three Line Break), popisovaná i v knize od Turka. [14] Zákazník tuto strategii předkládá s jinými parametry a bude vyhodnocena jeho profitabilita. V případě, že bude profitabilní bude tato metoda zákazníkem dále vyvíjena a optimalizována.

Základní parametry AOS tedy byly zákazníkem dodány v následující podobě:

- Analyzovat se budou pouze historická data.
- Systém bude obchodovat na majoritních párech (EURUSD, GBPUSD).
- Time Frame (TF) bude zvolen až podle ziskovosti, předpokládá se H1 nebo H4.
- Budou hledány 3 za sebou jdoucí svíčky stejným směrem.
- První podmínkou je, že všechny LOW a HIGH ceny těchto svíček budou mít rostoucí nebo klesající charakter podle směru těchto tří svíček.
- Druhou podmínkou je, že všechny OPEN a CLOSE ceny těchto svíček budou mít rostoucí nebo klesající charakter podle směru těchto tří svíček.
- Třetí podmínkou je, že mezera mezi OPEN cenou první svíčky a CLOSE cenou třetí svíčky bude mít určitý GAP, tedy cenovou mezeru určité velikosti. V základu je navržena cena 0,0050 jako nutný minimální GAP, který je předpokládán na TF H1.

- Zákazník požaduje také mít možnost omezení maximálního počtu naráz otevřených obchodů v obou směrech. Tedy jak LONG tak SHORT.

Při diskusi nad problémem přesného nastavení AOS bylo zjištěno, že zákazník nemá konkrétní představu o kvalitě tohoto systému. Z toho vyplynul nový požadavek zákazníka na nutnost vytvoření takového systému pro prvotní analýzu a případnou optimalizaci AOS, jenž systém správně nastaví pro aktuální pozici na trzích.

Základní parametry pro realizaci testovacího a optimalizačního prostředí dané zákazníkem:

- Vytvořit takový systém, který umožní analýzu a optimalizaci burzovních obchodních systémů, které bude možné samostatně programovat.

Zároveň byl vznesen dotaz, zdali je možné vytvořit takový prediktivní systém cenových pohybů, který by umožnil s dostatečnou přesností predikovat vývoj cen. Toto nebylo možné v daném okamžiku rozhodnout, proto byl domluven pouze dílčí požadavek na vytvoření tohoto systému, který by umožňoval s použitím umělé inteligence predikovat cenové pohyby s dostatečnou přesností alespoň padesáti procent. Tento požadavek byl pro mě výzvou pro využití nabytých znalostí studií na VUT Brno.

Dalším cílem tedy bylo pokusit se takovýto predikační systém naprogramovat.

Na základě svých osobních zkušeností s jedenáctiletým obchodováním na burze, kdy jsem pracoval na vývoji automatických obchodních systémů jak na zakázku, tak pro osobní potřebu, jsem byl schopen posoudit jak relevantnost požadavků zákazníka, tak i možnosti realizace. Zakázka tedy byla přijata, avšak s výhradou neurčitosti úspěšného řešení predikování vývoje cen s použitím umělé inteligence.

### **3.2 Strategie postupu vývoje automatického obchodního systému**

Vývoj strategie postupu vývoje automatického obchodního systému jsem rozdělil do pěti hlavních částí, které budou v dalších podkapitolách podrobně popsány:

- Zachycení myšlenky AOS
- hrubý vývoj
- upřesnění parametrů
- optimalizace
- analýza dat.

Touto posloupností jsem se také řídil při programování výše uvedené strategie dodané zákazníkem.

#### **3.2.1 Zachycení myšlenky AOS**

V této části si trader utvoří jakoukoli strategii, která jemu dává smysl, nebo ji vidí jen náznakově na cenových grafech konkrétních měnových párů a Time Frame. Případně svou myšlenku ještě zkombinuje s indikátory, jako jsou Fibonacciho klíčové hladiny rozložení cen, Support Resist hladiny, Moving Average křivky a jiné. [2] Jde o

navození situace, kdy trader najde potencionální pattern, který by šel obchodovat a má konkrétní potenciál opakovatelnosti. Například dlouhá svíčka pohlcuje malou svíčku, tři za sebou jdoucí svíčky v určité kombinaci, mnoho malých svíček a najednou jedna velká a mnoho dalších patternů, jejichž počet je prakticky nekonečný. V této fázi o strategii nevíme vůbec nic, tedy zdali bude zisková, kolik obchodů udělá, či jaké parametry má mít. Jde o situaci, která ovšem umožní tuto myšlenku zkusit naprogramovat do automatického obchodního systému.

### 3.2.2 Hrubý vývoj

Jedná se o naprogramování AOS uvažovanou strategií podle odhadovaných údajů, které byly získány pozorováním grafů, nebo jiným obdobným způsobem. Zde jde jen o správně naprogramovaný kód AOS, který by se mohl použít k testům. V této fázi se už rovněž při programování AOS nastaví nahrubo parametry tohoto systému, aby šla vyzkoušet jeho funkčnost. Pokud však za této situace již bude zřejmé, že uvažovaný vývoj nenastává, nebo má velmi malý ziskový potenciál, můžeme rovnou ukončit hlavní rozvoj myšlenky a případně včas provést její korekci. Tato situace významně ulehčuje práci na AOS a především výrazně šetří čas na rozvoj myšlenky, která minimálně podle prvotních náhledů chování AOS nefunguje. Neznamená to ovšem, že je daná myšlenka automaticky špatná, jen aktuální požadavky a poznání nevedenou k vytyčenému cíli, kterým je generování potencionálního zisku. V tento okamžik je případně nutné kontaktovat zadavatele, aby došlo ke korekci AOS.

### 3.2.3 Upřesnění parametrů

Tato část je úzce provázaná s předchozí částí vývoje, kdy se po tvorbě AOS v hrubém vývoji doladují patřičné parametry. [4] Například bychom chtěli použít rozsah GAP alespoň třicet základních cenových jednotek PIP, ale tato strategie s tímto nastavením negeneruje žádný obchod nebo minimum obchodů. Pak tedy musí dojít k doladění parametrů tak, aby vznikl dostatečný počet vzorků obchodů, které bude možné analyzovat v další části vývoje AOS. Zde také dochází k určení, zdali tento AOS bude možné použít i pro jiné páry, nebo je určen jen pro jediný konkrétní pár. Testují se rovněž odlehle hodnoty vstupních parametrů AOS, aby šlo určit, které se použijí pro optimalizaci. Tato analýza výrazně šetří čas nutný na optimalizaci snížením možných variant vstupů těchto odlehlých hodnot.

### 3.2.4 Optimalizace

Jedná se o stěžejní část pro maximalizaci zisku a minimalizaci rizika. Tedy nastavení RRR ideálně podle požadavků tradera. Zde se již nasazuje odladěný AOS po výše uvedených krocích a jsou nastaveny rozsahy patřičných vstupních parametrů s kroky v tomto rozsahu. Například při optimalizaci velikosti GAP víme, že předpokládaná ideální hodnota je třicet a odlehle hodnoty jsou deset a šedesát, nastavíme velikost vstupních parametrů pro optimalizaci deset až šedesát s krokem deset.

Nyní je spuštěna tato optimalizace na určených historických datech. Výsledkem optimalizace je tabulka  $1 \times 1 \times 1$ , kterou si lze představit jako 3D graf s parametry  $x$ ,  $y$ ,  $z$ , ale zobrazený v 2D. Tato tabulka umožňuje vidět chování konkrétních vstupních parametrů tohoto AOS. Následně je s těmito daty provedena analýza, zacílená na to, v jaké oblasti vstupů by se mělo AOS pohybovat. Hledáme shluky, například tmavě zelených pásů či čtverců okolo zvolených parametrů. Pokud tato oblast neexistuje, musí se zvolit optimalizační testy s jinými rozsahy parametrů, či jinými vstupními proměnnými. V horším případě půjde o náhodné generování zisku se zcela specifickými parametry, tudíž nepůjde o nalezení AOS, který bude fungovat i na jiných předem neznámých datech.

### 3.2.5 Analýza dat

Cílem analýzy dat je vyhledávání shluků ziskových obchodů a jsou vybírány i takové obchody, které nejsou nejziskovější, ale umožňují nám na těchto oblastech určit daleko vhodnější vstupní parametry, neboť odchylka od vstupních hodnot nezpůsobí kolaps AOS. Parametry vstupních dat středu této oblasti by se měly stát předpokládaným ideálním nastavením pro AOS.

## 3.3 Vývoj testovací části systému

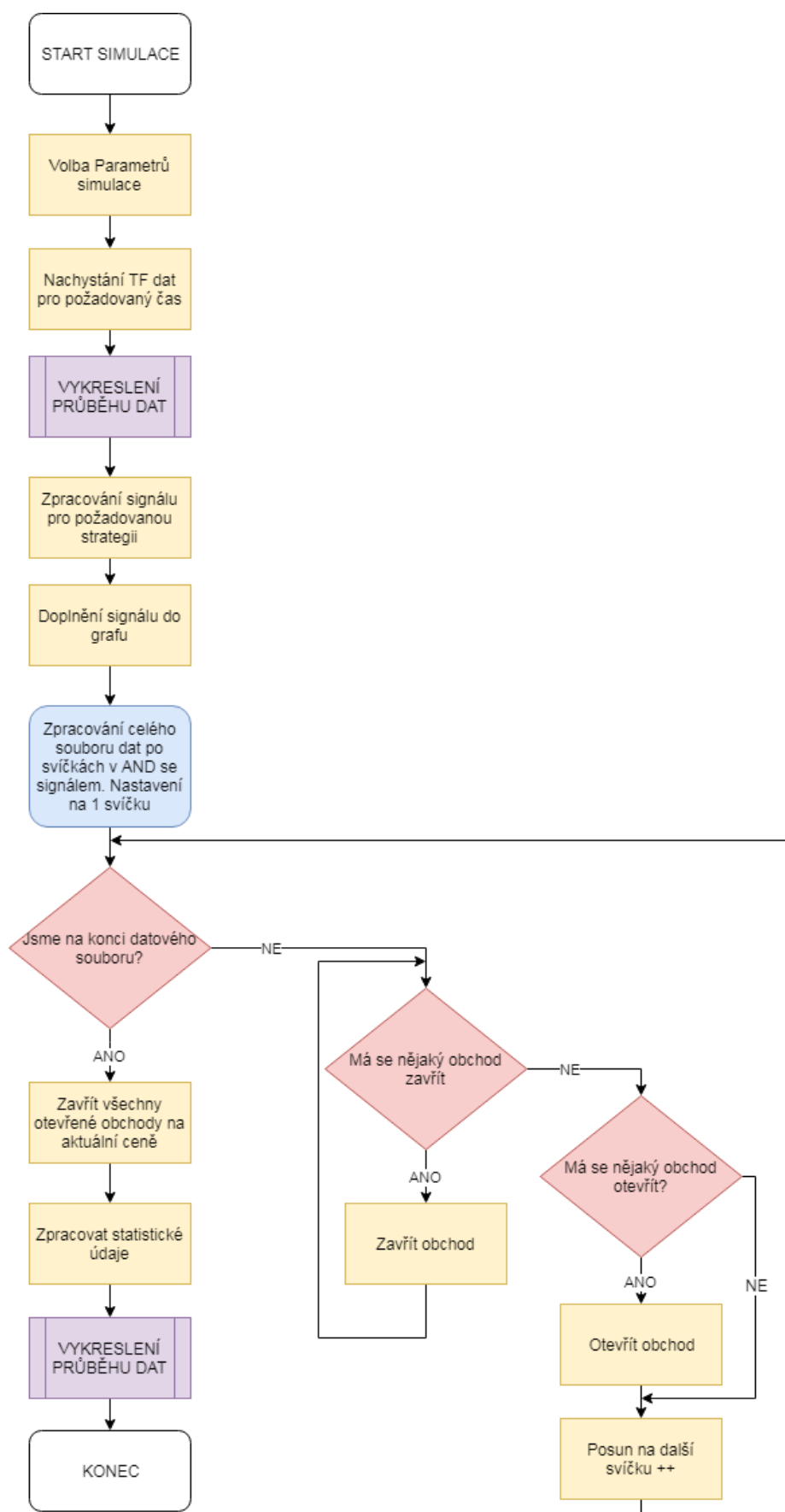
Vývoj se bude skládat z několika samostatných modulů, které jsou spolu navzájem v interakci. Jejich vývoj není náhodný a je třeba zachovat určitou posloupnost kroků, aby šly realizovat kroky navazující. Celý proces průběhu činnosti testovacího a simulačního systému je přehledně zachycen ve Flow Chart grafech vložených ke každé podkapitole s popisy modulů, které zachycují celý proces. Jedná se o tyto hlavní moduly:

- Modul simulátoru
- modul optimalizátoru
- modul AI predikce
- modul testy AOS s predikcí
- modul vykreslení průběhu dat
- modul vykreslení  $1 \times 1 \times 1$  grafu.

Nejdříve bylo nutné vytvořit modul simulátoru. Tento modul je zobrazen ve Flow Chart grafu na obrázku níže. Tento graf přehledně zobrazuje nezbytný postup v procesu chodu celého simulátoru včetně rozhodovacích podmínek. Lze použít jak samostatně, tak i posléze pro modul optimalizátoru. Tato část prakticky zpracuje signál do takové podoby, jako by byl reálně obchodovaný.

Jednotlivé kroky, které vyžadují jistou konverzi dat nebo informaci o průběhu, jsou zobrazeny v oranžových obdélnících. Části ve fialovém dvojitém orámování znázorňují spouštění modulu nebo podprogramu, například vykreslování průběhu dat samostatným modulem. Modrý obdélník obsahuje informaci o cyklickém zpracovávání

určitého bloku dat obvykle posouvaný po jednom kroku. Aby bylo možno zachytit i rozhodovací proces, jsou použity rozhodovací kosočtverce, pomocí kterých je program řízen jakožto rozhodovací podmínka. Například zdali se má nějaký obchod zavřít, otevřít a podobně. Bílé obdélníky jsou čistě informativní a mají oddělovací a popisný charakter.



Obrázek 4 Flow Chart modulu simulátoru [vlastní zpracování]

### 3.3.1 Zajišťování TICK dat

Pro zajištění simulace je třeba nejdříve získat co nejkvalitnější a nejjemnější historická data. Tato data se podařilo získat z veřejně dostupného zdroje a jedná se o data vygenerovaná firmou MetaQuotes pocházející z platformy MT4 a MT5. Data jsou dostupná na webu histdata.com. [13]

Tato data jsou k dispozici ve formátu CSV a obsahují až devatenáctiletou historii. Z tohoto důvodu i vytvářený systém počítá maximálně s kapacitou dvacetiletého zpracování dat. Zmiňovaná data jsou ovšem k dispozici pouze jako M1 TF a nejsou tedy k dispozici přesná TICK data, která by ukazovala vývoj ceny v rámci jedné minuty. Data nejsou k dispozici z toho důvodu, že firma MetaQuotes neposkytuje menší TF než jednu minutu. Přestože data nemáme k dispozici, je možné pracovat s dostatečnou přesností simulace, což je dostačující pro otestování jakéhokoli AOS pracujícího na alespoň M5 TF. Je tedy zřejmé, že čím větší TF, tím méně bude toto omezení vadit. V rámci simulace bylo nutné i tuto situaci vyřešit. Pokud tedy nemáme TICK data, používáme hodnoty stejné jako nejmenší dostupný TF. Zde tedy bude použit M1 TF. Není to ideální postup, ale kvalitnější data k dispozici nejsou, a to ani v prostředí MT4. Při kolizi na dosažení Target Profitu (TP) a Stop Loss (SL) na jedné TICK svíčce bylo rozhodnuto vzít v úvahu raději ztrátový obchod než ziskový, aby strategie byla raději pesimističtější, než optimističtější. Výsledky jsou tedy vždy zkresleny určitou chybou v důsledku těchto umělých zásahů do stanovení koncových cen obchodů.

Tato data jsou v běžném kótování pro platformu MT4 a jsou tedy v časovém pásmu GMT + 5. Tato skutečnost je stejná, jako kdyby se obchodovalo na MT4, proto data nejsou nijak upravována.

Důležitost konverze z CSV dat do xxx.mat souborů je velice dobře viditelná při zpracování datového souboru EURUSD z roku 2018, který čistě v textové podobě obsahuje data o velikosti 20014 kB a obsahuje 372607 minutových svíček. Po konverzi do matového souboru se podařilo uložit tato data do proměnné, jejichž velikost je již jen 5941 kB. Dochází tedy k významné úspoře paměti při práci s rozsáhlými daty. Ukazuje to, že ukládací forma souboru s proměnnými v programu MATLAB do xxx.mat souborů je výrazně optimalizovanější a úspornější. Této vlastnosti je v mém programu využíváno.

Samotná konverze zabere nějaký strojový čas a ukázka zpracování datového souboru EURUSD rok 2018 je zobrazena na výpisu z MATLABu:

```
start konverze
```

```
Pocet RAW dat: 372607
```

```
zkonvertovan TF M5 za: 6.284
```

```
zkonvertovan TF M15 za: 1.9813
```

zkonvertovan TF M30 za: 1.0132

zkonvertovan TF H1 za: 0.54448

zkonvertovan TF H4 za: 0.19814

zkonvertovan TF D1 za: 0.048463

zkonvertovan TF W1 za: 0.0515

zkonvertovan TF MN za: 0.030467

vsechny TF konvertovany za: 10.1515

stop konverze

Zde vidíme, že celková konverze dat trvá 10,15 sekund a výrazně se doba zpracování snižuje s vyšším TF. Toho jsem docílil logickou úvahou, že pro generování vyššího TF je plně dostačující znát hodnoty TF o jedno nižší, neboť již obsahují všechna potřebná data i z menších TF. Například TF H1 je generován z TF M30. Je tedy vidět, že celá konverze trvá už pouze půl sekundy namísto očekávaných šesti sekund, tak jako na TF M5. Touto metodou si můžeme dovolit konvertovat i opravdu velká data zachycující desetiletý vývoj cen na burze a celá konverze zabere okolo sto sekund. Pokud by tato metoda nebyla použita, čas potřebný pro konverzi by byl okolo pětiset sekund. Již zde se ukazuje potřeba optimalizace dat a navržení optimalizačních procesů při jejich zpracovávání.

### 3.3.2 Ukládání dat

Bylo nutné vyřešit, jakým způsobem bude s daty pracováno v průběhu chodu programu. Zdali použít datový sklad nebo jít cestou jiné formy ukládání. Vzhledem k možnostem MATLAB jsem zvolil metodu ukládání do třídy a pro tento účel jsem si vytvořil třídu classData. Je to dáno tím, že v této třídě jsou jednak uložena samotná data jako atributy, ale také tato třída obsahuje patřičné metody, které s těmito daty dokáží pracovat přímo. Tím bylo docíleno zrychlení přístupu k datům právě pomocí metod a zajištěna i nemožnost jejich poškození vlivem zapouzdření. Přístup k této třídě je tedy řešen zcela objektově a umožňuje vysoce přehledný zápis kódu, který bude důležitý právě při tvorbě jiných AOS.

Tím bylo docíleno také toho, že konverze na jiný zcela nestandardní TF například M7, M13, H9, D3 je možné provést prakticky dvěma řádky kódu. Využitelnost těchto TF v daném okamžiku není, ale jsou známy metody, kdy jsou i takto bizarní TF používány k zvláštní predikci. Tato funkčnost ovšem je prezentována jako nadstavbová, doplňková záležitost.

Příklad konverze dat z M1 na M5 je ukázáno na tomto kódu.



```
Mx = data.convertDataToMx(5); %zavolani funkce na konverzi TF  
dataM5.setData(Mx); %ulozeni ziskaneho TF do classData
```

Rychlost zpracování těchto převodů velmi ulehčuje právě vytváření programu v MATLABu, který je znám pro svoji vynikající a rychlou práci s maticemi. Na vzorku 6000 svíček M1 se při konverzi na M5 získá 1779 M5 svíček, ale vyčištěním se zničí 576 špatných svíček. Jde především o GAP mimo obchodní hodiny. Tímto čištěním získáme podstatně lépe optimalizovaná data pro vyšší TF. Z toho vyplývá jedna velmi nepříjemná vlastnost, kdy program nedokáže nabídnout M1 data pro obchodování i když nám nic nebrání je využít.

U konverze je velmi důležitou vlastností to, aby data mohla být zcela nezávislá na konverzi. Pokud tedy data začínají v čase 16:59 místo v 17:00 svíčka je správně přepočítána jako svíčka předchozího času 17:00 a začíná tedy správně na 16:55 v případě M5 dat. I kdyby tyto časové údaje byly nelogické, přesto jsou to naše diskrétní vstupní data, se kterými pracujeme. Proto je zmiňována důležitost ošetření těchto anomálií. Zvláště u dat z jiných časových pásem není známo, kdy jsou data hypoteticky správná nebo kde bychom s nimi v tomto čase již ani nepočítali. Výhodnější je konverze velkých časových TF jako den, týden, měsíc z důvodu zahrnutí svíček blíže předpokládanému intervalu, což znamená méně osamocených dat.

V následujícím obrázku je názorně předvedeno rozdělení TF M1 červenými dělicími čarami pro interval TF M5. Z těchto hodnot jsou nové svíčky M5 vypočítávány tak, že je použita OPEN cena první svíčky a CLOSE cena poslední svíčky a spočítány maximální a minimální hodnoty jako HIGH a LOW hodnoty pro M5 interval. Vznikne tedy nová M5 svíčka s hodnotami OPEN rovná se 1.2004, CLOSE 1.2005, HIGH bude mít hodnotu 1.2010 a LOW hodnotu 1.2002.

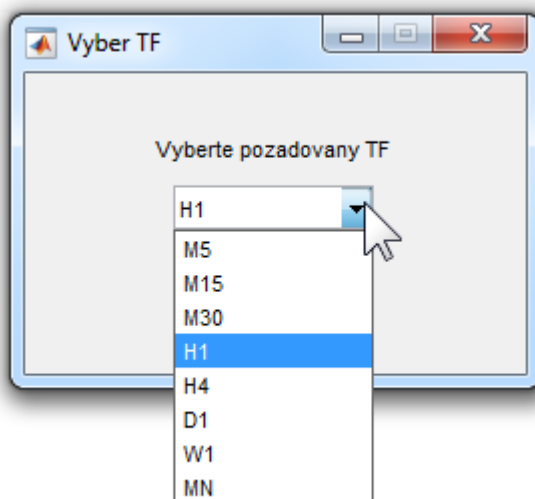
	Time	1 Open	2 High	3 Low	4 Close	5 Volume
1	2018-01-01 17:00	1.2004	1.2010	1.2004	1.2010	0
2	2018-01-01 17:01	1.2008	1.2010	1.2002	1.2003	0
3	2018-01-01 17:02	1.2004	1.2004	1.2004	1.2004	0
4	2018-01-01 17:03	1.2004	1.2005	1.2003	1.2005	0
5	2018-01-01 17:04	1.2005	1.2005	1.2005	1.2005	0
6	2018-01-01 17:05	1.2005	1.2005	1.2005	1.2005	0
7	2018-01-01 17:06	1.2005	1.2007	1.2003	1.2005	0
8	2018-01-01 17:07	1.2005	1.2005	1.2003	1.2005	0
9	2018-01-01 17:08	1.2005	1.2009	1.2005	1.2009	0
10	2018-01-01 17:09	1.2008	1.2010	1.2008	1.2009	0
11	2018-01-01 17:10	1.2009	1.2010	1.2008	1.2008	0
12	2018-01-01 17:11	1.2008	1.2009	1.2008	1.2009	0
13	2018-01-01 17:12	1.2009	1.2009	1.2008	1.2008	0
14	2018-01-01 17:13	1.2008	1.2008	1.2008	1.2008	0
15	2018-01-01 17:14	1.2008	1.2009	1.2008	1.2009	0
16	2018-01-01 17:15	1.2009	1.2011	1.2004	1.2004	0
17	2018-01-01 17:16	1.2004	1.2004	1.2004	1.2004	0
18	2018-01-01 17:17	1.2004	1.2004	1.2004	1.2004	0
19	2018-01-01 17:18	1.2004	1.2005	1.2004	1.2005	0
20	2018-01-01 17:19	1.2004	1.2005	1.2004	1.2005	0
21	2018-01-01 17:20	1.2005	1.2005	1.2004	1.2005	0
22	2018-01-01 17:22	1.2005	1.2005	1.2005	1.2005	0
23	2018-01-01 17:23	1.2005	1.2006	1.2005	1.2006	0
24	2018-01-01 17:24	1.2006	1.2006	1.2005	1.2005	0
25	2018-01-01 17:25	1.2005	1.2005	1.2005	1.2005	0

Obrázek 5 Konverze M1 svíček do M5 rozdělení [vlastní zpracování]

Výjimka vzniká při zpracování posledních dvou TF, kdy konverze na týdenní a měsíční data jsou kvůli jinému rozptylu času nutná zpracovat jinou metodou ve třídě `classData`. Zde totiž nesouhlasí počáteční hodnoty obchodu a daného intervalu. Všechny ostatní metody jsou totiž zpracovávány jako minutové.

### 3.3.3 Vytvoření modulu pro výběr testovacích dat

V tomto modulu je řešena situace, kdy trader si vybírá konkrétní měnový pár z `xxx.mat` souborů vytvořeného pomocí předchozí konverze dat. Následně si v uživatelském okně volí požadovaný TF. Tato volba je společná pro celý další proces testování v navrhovaném systému. Jde tedy o to, aby si uživatel dopředu vybral, na kterém TF bude probíhat test. Výběr jiného TF způsobí anulování výsledků a celý proces testování se zopakuje s nově zvoleným měnovým párem a TF. Na výběr má uživatel z osmi standardních TF od M5 představující pěti minutovou svíčku po MN představující měsíční svíčku. Nejčastěji je předpokládán test na zvýrazněném TF H1 představující hodinová data.



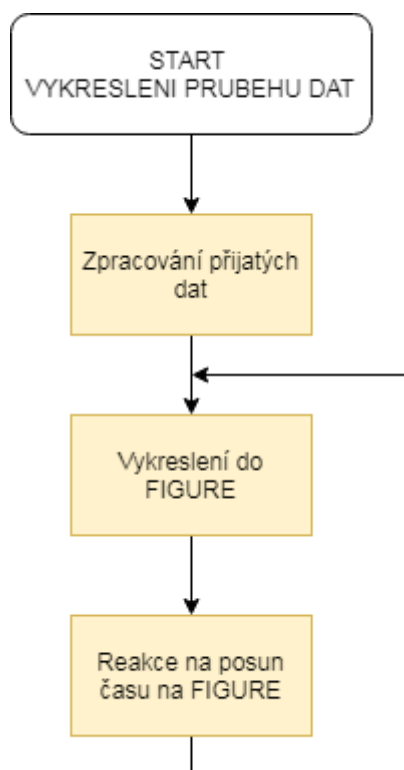
Obrázek 6 Uživatelské okno s výběrem TF [vlastní zpracování]

Posuvníky na grafech umožňující pohyb zobrazení časových svíček jsou přizpůsobené až do hodnot deseti milionů těchto svíček. To odpovídá při M1 datech cca dvaceti šesti letům záznamů. Program je však záměrně omezen a počítá s využitím dat v rozsahu pěti milionů jako největšího extrému při testech za období celých deseti let. Toto omezení nevylučuje z funkčního omezení softwaru, ale ze skutečnosti, že data, která jsou starší než deset let, jsou velmi historicky vzdálená a nemá tudíž smysl se jimi dále zabývat při testování AOS reagujícího na aktuální vývoj cen. Testeři AOS běžně testují svá data za období pěti let, a to ještě při velkých TF, jako jsou H4 či D1 svíčky. Při používání H1 svíček se již tato doba hodně zkracuje a to na období maximálně dvou až tří let právě pro svoji velkou časovou náročnost na pomalých časových testerech, jako je tester MT4. Toto omezení se nás sice přímo netýká, ale rozsah testů je využíván i v tomto mém systému.

Při tvorbě těchto dialogových oken jsou využita vlastní dialogová okna vytvořená v MATLAB, jak je vidět na výše uvedeném obrázku. Jsou rovněž využívána dialogová okna přímo od operačního systému Windows, například pro načítání a ukládání souborů. Tím je dosaženo standardizovanějšího vzhledu a uživatel tak pracuje ve známém prostředí.

### 3.3.4 Vytvoření modulu pro zobrazování pomocí svíček

V níže vloženém obrázku jsou zobrazeny svíčky měnového páru EURUSD na TF H1 od začátku roku 2018. Jednotlivé svíčky představují právě jeden interval H1. Pomocí tohoto zobrazení si tester zajistí přehled o vývoji cen na jím zvoleném TF a může si strategii promýšlet na libovolně zvoleném časovém úseku. Toto je hojně využíváno právě u techniky Price Action, která reaguje jen na cenové pohyby a jejich vyjádření pomocí svíček konkrétního TF. Samotná strategie prvotního pohledu při vytváření AOS bude popsána v části zabývající se implementací AOS do mnou vytvářeného systému.



Obrázek 7 Flow Chart vykreslování průběhu dat [vlastní zpracování]



Obrázek 8 Graf svíček EURUSD H1 [vlastní zpracování]

Zpracování dat s vykreslením do okna není časově náročné na běžných velikostech zobrazení a pohybuje se řádově v jedné sekundě pro viditelné zobrazení svíček. Zvětšováním rozsahu zobrazení až na úroveň přehledu dlouhodobého trendu se úroveň vykreslování logicky zvyšuje, ale i přesto vykreslení celého ročního průběhu trvá asi dvacet sekund. Toto zobrazení však není pro tradera smysluplné, neboť svíčky H1 v takto obrovském rozsahu nebudou viditelné na žádném testovacím softwaru. Proto si trader přepne na vyšší TF, například denní a zobrazení tři sta třinácti svíček již bude

dávat podstatně čistší obraz cenových pohybů a jeho zobrazení trvá pouze jednu a půl sekundy. Na rychlost vykreslování má i zásadní vliv informace o tom, že jsou vykreslovány jen svíčky, které máme k dispozici. Pokud v daném čase svíčka nemá okótovaná historická data, nejsou vykreslována.

Při vykreslování je také potřeba ošetřit malý počet svíček u velkých TF. Toto přenastavení zobrazení se děje zcela automaticky a reaguje na celkový počet zobrazovaných svíček. Dochází taktéž k úpravě rozsahu posuvníků pod grafem.

### 3.3.5 Zpracování signálu pro požadovanou strategii

Aby bylo možné zpracovat signál, je dobré připravit si data se všemi potřebnými parametry. V mém případě byla použita TimeTable tabulka, se kterou umí MATLAB výborně pracovat. Vytváření samotného signálu dopředu je velice výhodné právě kvůli ušetření času při samotném hrubém vývoji strategie. Nedochází při něm totiž k obchodování, ale z výstupních tabulek jsme okamžitě schopni rozhodnout, zdali toto nastavení odpovídá alespoň očekávaným výsledkům.

Pro zpracování signálu, je třeba vytvořit modul, který bude umět zpracovávat svíčky z konkrétního TF. Jednotlivé svíčky jsou procházeny postupně a na nich je vyhodnocována shoda vstupních parametrů AOS s patternem historických dat.

K dosažení vysoké rychlosti tvorby dat signálu bylo nutné uvažovat v programu o takových procesech, které umožní optimalizaci chodu vyhodnocování vstupních podmínek AOS. Základní myšlenka, na kterou upozorňuje MATLAB při samotném psaní kódu je ošetření stavu při ukládání proměnné. Při ukládání do této proměnné, jakožto matice, by docházelo k tomu, že by se neustále tato proměnná přesouvala v paměti. Tím by docházelo k neúměrnému zpomalování celé tvorby signálu. Ošetření této situace lze zajistit alokováním proměnné samými nulami. Na základě toho dojde k připravení paměťového místa dostatečně dlouho dopředu a ukládání do této proměnné probíhá mnohonásobně rychleji. Je to jedna z hlavních optimalizačních metod, která výrazně pomáhá rychlosti zpracování kódu.

Další vysoce významnou skutečností při optimalizaci rychlosti kódu je využití významné výhody MATLABu při jeho vyhodnocování AND logických spojení. MATLAB jako jazyk čtvrté generace jde cestou, kde nevyhodnocuje nejdříve všechny výrazy spojené spojkou AND uvnitř výrazu, který by jako celek vyhodnocoval až na konci. Vyhodnocuje postupně celý výraz směrem zleva doprava. Pokud je v tomto pořadí zleva první podmínka nepravdivá, okamžitě ukončuje podmínku jako nesplněnou. Tím šetří strojový čas nevyhodnocováním všech podmínek, které by samy o sobě neovlivnily výsledné vyhodnocení výrazu.

Tento způsob vyhodnocování umožňuje optimalizovat vstupní podmínku AOS signálu pomocí AND spojení a provádět test v rámci jednoho jediného řádku. Vyberáme takové podmínky, aby výraz první zleva měl největší váhu. V běžném programovacím jazyce toho není možné dosáhnout, jelikož by se muselo použít sériově několik

testovacích podmínek. Ukázka mého zpracování vstupní testovací podmínky dodané zákazníkem je zobrazené na kódu z MATLABu níže.

```
%hlavni test formace 3 svicek vse je && ukoncuje se okamzite
%jakmile neni splnena jakakoli podminka odleva, tim jde o
% velmi rychly test !!!
if (d.Open(1) < d.Open(2)) && (d.Open(2) < d.Open(3)) && ...
    (d.Close(1) < d.Close(2)) && (d.Close(2) < d.Close(3)) && ...
    (d.High(1) < d.High(2)) && (d.High(2) < d.High(3))
    gap = d.Close(3) - d.Open(1); %zjistime si GAP
    if gap >= nutnyGap %pokud je gap splnen muzeme zadat obchod
```

Na konci je pouze druhá podmínka IF, která již pouze zpřehledňuje zápis kódu. Mohla by být také v celém výrazu, ale zde byla upřednostněna přehlednost před rychlostí. Doplňováním dalších podmínek, nebo upravováním hlavního výrazu bude určeno, jakým způsobem bude AOS reagovat na historická data. Pro objasnění této strategie jsem upravil na obrázku níže zobrazení svíčkového grafu tak, aby bylo zřetelné, v jakém okamžiku je signál zpracován a kdy již není vytvářen.

Na tomto obrázku vidíme pět potencionálních obchodů s možností jejich vstupu. Všechny tyto obchody splňují všechny podmínky dodané zákazníkem s tím rozdílem, že se liší pouze ve velikosti GAP hodnoty. Tato GAP hodnota rozhodne, jestli tento signál bude zpracován, nebo nebude. Obchod s označením čísla jedna, tři a deset nebude zaznamenán jako obchodní signál, jelikož tuto podmínku s GAP hodnotou nesplňuje. Naopak obchody číslo pět a sedm splní úplně všechny podmínky včetně velikosti GAPU a budou vygenerována data s možností jejich obchodování.



Obrázek 9 Ukázka zobrazení vstupů AOS[vlastní zpracování]

Třetí optimalizační podmínkou je využití další přednosti MATLABu, kdy při parametrickém předávání proměnné zprvu nedochází k fyzickému kopírování této proměnné, ale dojde k předání odkazem. Ve chvíli, kdy je do této předávané proměnné proveden nějaký zápis, je až nyní vytvořena lokální kopie. To tedy znamená, že si

můžeme dovolit i předávání velkých bloků dat, řádově dvaceti nebo třiceti MB, aniž by docházelo ke kopiím v rámci paměti. Stačí totiž dodržet podmínku, že se z této předávané proměnné bude pouze číst, ale ukládat se bude do jiné proměnné.

Modul, který zajišťuje zpracování signálu, je zcela variabilní a neomezuje se pouze na techniku Price Action. Je možné využít implementované indikátory, které jsou již součástí MATLABu například MA indikátor. Není žádný problém naprogramovat si vlastní indikátory. Bude to však znamenat zpomalení generování sady signálů. V mém případě i v požadavku zákazníka není o tyto indikátory zájem, tudíž nejsou ani implementovány v současné verzi testovacího programu. To by znamenalo přidání samostatných modulů pro konkrétní indikátory, které by byly spouštěny v rámci hlavní testovací podmínky při generování signálu. Nyní uvedu hlavní spouštěcí smyčku, která provádí zpracování vstupního zadání AOS do signálu s využitím výše zmíněných postupů.

```
for x = varaTP
    for y = varaSL
        for z = varaGap
            cisloTestu = cisloTestu + 1;
            AOS_TLBKS(x,y,z,startSim,konecSim);
            pocetSig = size(signal); %ulozime pocet signalu
            profitSig = 0; %profit
            %maxObchodu = 0; %maximum obchodu
            nazevDat = filename; %nazev souboru dat
            castF = TF; %nazev TF
            ts(cisloTestu) = struct('sTP',x,'sSL',y,'sGap',z,...

'sPocet',pocetSig(1,1),'sProfit',profitSig,'sMaxObch',maxObchodu,...
'sNazev',nazevDat,'sTF',castF,'sCT',cisloTestu,...
'sSignal',signal);
            disp(['Signal na TF ',string(castF),'poradi cislo: '
num2str(cisloTestu)]) %nutno prevest na text
        end
    end
end
```

Při generování signálů lze proces přiblížit příkladu zpracování mnoha signálů na měnovém páru EURUSD s TF H1, kde sada dat obsahuje 6226 svíček. V případě, že je strategie navržena pro dvacet osm obchodů, bude zpracován signál pro všechny obchody za necelých osmdesát sekund. Je třeba vzít v úvahu, že procedura zpracování signálů je časově nejnáročnější částí testeru. Proto je nutné dbát tak významně na optimalizaci kódu. Vzhledem k tomu, že nemá význam vytvářet dva různé prakticky totožné moduly, je modul pro optimalizaci kódu AOS využit i jako simulátor. Pokud je modulu na tvorbu signálů dodán jen jeden parametr místo sady vstupních parametrů, pracuje tento modul jako tvůrce jedné sady signálů. Rychlost se nepatrně zhorší, což je v případě generování pouze jedné sady zanedbatelný nedostatek.

Vzhledem k tomu, že spíše upřednostníme rychlost před jednoduchostí zadávání vstupních dat, je použita pro zadávání dat jednotková cena odpovídající hodnotě jednoho PIP. V případě majoritních párů GBPUSD, AUDUSD, USDCHF jsou tyto ceny jednotné a představují 0,0001 ceny jakožto jednoho PIP. Při použití měnových párů, kde se vyskytuje japonská měna pod zkratkou JPY, počítáme s hodnotou jednoho PIP na úrovni 0,01 ceny. Známe-li tedy hodnotu jednoho PIP, není problém nastavit si odpovídající rozsahy při zadávání vstupních parametrů. Pokud tedy potřebujeme docílit například GAP třiceti PIP, budeme toto rozpětí zadávat jako hodnotu odpovídající 0,0030 ceny. V případě japonských párů by se jednalo o hodnotu odpovídající 0,30 ceny. Záměrně nepíši jaké ceny, neboť pro tester je to nepodstatné a každý tester má tuto cenu jinou. Někde bude měnou EUR, jinde USD a podobně.

Při zadávání obchodů není brán v potaz výpočet margin hodnoty, neboť je toto nastavení opět odlišné pro všechny obchodní platformy. Pro samotné testování AOS není podstatné, jak je vysoký tento margin, neboť neovlivní výslednou testovanou strategii.

Lze tedy konstatovat, že testování probíhá na neutrální měně bez určené hodnoty PIP. Bereme tedy jen číselné vyjádření ceny dle výše uvedených přepočtů. Jde o mnohem jednodušší pojetí testů, ovšem zcela univerzální a bez vazby na jakoukoli měnu.

Jelikož není dopředu možné určit, kolik času bude třeba na výrobu signálů, je v programu zařazen mechanismus, který informuje uživatele o průběhu zpracovávání datového souboru. Po každých pěti procentech zpracovaného datového souboru je uživateli zobrazeno hlášení, které ho informuje, v kolika procentech zpracovaného datového souboru se nachází spuštěný proces. I když se jedná o zdržení, jde o čas cca šedesát milisekund navíc.

### 3.4 Vývoj simulační části systému

V tomto modulu se provádí vyhodnocování výše uvedených získaných signálů. Tyto signály jsou procházeny postupně tak, jak je uvedeno na Flow Chart grafu simulace. Jednotlivé signály tak jak přicházejí, jsou zpracovány tímto modulem a rozhoduje se o potřebě zavřít nějaký otevřený obchod a případně se volí otevření nového obchodu. Tento proces má přednastavené pořadí, kdy je zásadní nejdříve na konkrétní historické svíčce provést test onoho zavření a provést případně i zavření tohoto obchodu. Není podstatné, jestli se tento obchod zavře ve ztrátě nebo v zisku. Po detekci všech otevřených obchodů, které by se mohly potenciálně zavřít, se teprve přistupuje k onomu zjištění nutnosti otevírání nových obchodů dle přednastavených signálů.

Z tohoto důvodu není možné testovat signály až na konkrétní svíčce, kde se má zadávat obchod, ale je třeba procházet postupně celý datový soubor historických dat od začátku až do konce. Na jiných svíčkách totiž může dojít k potřebě některý z dříve otevřených obchodů zavřít.



Jak vyplývá i ze zadání zákazníka, je simulátor obchodování realizován tak, že se vždy otevírá jen jediný obchod na jakékoli svíčce. Tento záměr usnadňuje i tvorbu AOS, který tak nemusí zohlednit podmínky, které panují při použití AOS přímo v platformě MT4. Je to sice jisté omezení pro určitý typ strategií, zároveň však byl tento požadavek shledán jako vysoce důležitý pro rychlost testů zejména při generování signálů optimalizace strategie AOS. Výsledný kód je tím značně kratší a nemůže dojít k multiplikování vstupních signálů na stejné svíčce. Zde je dobré zdůraznit i fakt, že platforma MT5 už s multiplikovanými vstupy nepočítá a nejdou v této platformě zadávat jako samostatné obchody. Toto omezení je považováno za významně větší problém pro mnoho traderů a také důvod, proč je tester vyvíjen prioritně pro platformu MT4. Omezení sdružených obchodů v MT5 i nadále zůstává a v testeru tato funkčnost není zahrnuta a ani není v plánu ji implementovat.

Přestože problém s rychlostí simulace obchodování není úplně kritický, je nezbytné zmínit použité řešení dřívějšího vytváření signálů, než jejich obchodování. Tato posloupnost zapříčiní rozdělení obchodování na dva různé moduly. Výsledná rychlost je podstatně vyšší, než kdyby se signál vytvářel až v průběhu obchodování. Pokud se na svíčce nemá otevřít žádný obchod, je ihned jedinou podmínkou proveden posun na další historická data. Rovněž toto rozdělení umožňuje spouštět simulaci obchodování na předem vytvořených signálových datech. Rozdělení signálních dat od vlastní simulace obchodování nám umožní spouštět simulaci kdykoli z dříve uložených dat signálů.

Ona nutnost dřívějšího zavření obchodu je použita i při vyhodnocování maximálních počtů naráz otevřených obchodů. Může se to jevit jako zdánlivá maličkost, ale vyhneme se situaci, kdy by bylo nutné otevřít mnoho souborů, aniž by to dovolovala výše obchodního účtu. Tato skutečnost významně ovlivňuje výkon AOS, je však nutné s tímto parametrem maximálního počtu otevřených obchodů počítat, neboť pouze trader ví, kolik obchodů si může dovolit obchodovat naráz při zvoleném riziku na konkrétním měnovém páru.

Například omezení pěti otevřených obchodů naráz neznamená, že trader nemůže více obchodů otevřít. Jde jen o omezení na jeden konkrétní měnový pár, který se testuje. Pokud trader obchoduje například čtyři měnové páry, je nutno počítat se skutečností, že může dojít naráz k otevření až dvaceti obchodů. Lze předpokládat, že takové množství obchodů už nebude možné v reálu zadat, neboť obchodní účet tradera mu pomocí marginu zabráni otevírání dalších obchodů. Z tohoto důvodu je třeba, aby trader ještě před testy měl představu o možnostech svého obchodního účtu. Proto je při testování AOS na tuto skutečnost kladen vysoký důraz prostřednictvím omezení maximálního počtu otevřených obchodů. Obchodování v tomto softwaru všechny signály, které by se měly otevřít nad tuto nastavenou maximální hranici, nekompromisně zahazují a nejsou obchodovány. Dochází tedy ke stejné situaci jako v reálném obchodování, kdy se do obchodu nevstupuje, pokud není na účtu dostatek finančních zdrojů na pokrytí marginu s náležitou rezervou.

Jsem si plně vědom, že tato skutečnost jak negativně, tak i pozitivně ovlivňuje testy AOS. Skutečnost, že nebyl signál zobchodován je zaznamenána v tabulce výsledků simulace obchodování a je možno provést i vyhodnocení strategie v případě, že je nutno zvednout hranici maximálních obchodů. Obvykle trader nedokáže předem odhadnout velikost omezení maximálního počtu obchodů. Z tohoto důvodu byla výchozí hodnota tohoto parametru nastavena na počet maximálně pěti naráz otevřených obchodů. V rámci testů lze tuto hodnotu samozřejmě změnit.

V této verzi simulátoru nejsou brány v potaz spready (odměna brokera za zprostředkování obchodu), jelikož nejsou předem známy a muselo by dojít k aktualizaci údajů přesně podle nastavení konkrétního poskytovatele obchodní platformy. Toto by však šlo použít pouze u Market Markera, který používá fixní spready. Mnoho brokerů především typu ECN používá spready variabilní a v tom případě by nastavení těchto fixních spreadů bylo chybné. Tato situaci by se tedy nedala vyřešit, jelikož se hodnota spreadu mění na každém TICK. To také považuji za hlavní důvod proč i tester zabudovaný v MT4 používá stejnou metodu nezahrnování cen BID a ASK a vyhodnocují se pouze BID ceny bez spreadu. Tato metoda vyhodnocování cen se nebude měnit ani v dalších verzích.

Poslední akce, kterou simulátor provádí na poslední svíčce, je zavření všech obchodů podle aktuální ceny CLOSE této poslední svíčky. K tomu dochází bez ohledu na to, v jakém stavu se otevřené obchody nacházejí. Možné jsou pouze dvě varianty, jak tuto situaci vyřešit, a to buď výše uvedeným způsobem na uzavírání otevřených obchodů na CLOSE ceně. Druhá varianta je anulovat všechny tyto otevřené obchody. Tester MT4 používá metodu CLOSE ceny, a proto také byla v mém testeru zvolena stejná strategie zavírání obchodu na poslední svíčce. Jedná se o situaci jako v případě ručního zavření všech obchodů samotným obchodníkem. Po konzultaci a vysvětlení situace zákazníkovi, je i jím tato použitá metoda považována za akceptovatelnou. Dochází samozřejmě k nepřesnostem při vyhodnocování úspěšnosti strategií, ale jedná se jen o maximálně jednotky otevřených obchodů, které jsou navíc zavírány předvídatelným způsobem. Při této metodě není zohledněno, zdali je obchod uzavřen v zisku nebo ve ztrátě.

### 3.5 Optimalizace

Modul optimalizace sdružuje oba předchozí moduly, které prováděly testování a simulaci. Oba jsou již předem připraveny, aby mohly provádět nejen testy na jediném nastavení, ale umí i zpracovat širší vstupní data. Tato vstupní data jsme již získali ze simulace s jedním nastavením od zákazníka. Prvotní hodnota se upřesňuje pomocí jednotlivých testů a hledá se hraniční hodnota využitelnosti AOS. Získáme dvě hladiny a tyto hladiny použijeme pro vstupní data našeho optimalizačního systému. Aby bylo možné tato data nasimulovat, musíme provést hromadné generování signálů pro všechna tato nastavení a posléze tyto získané signály dodat do modulu simulátoru obchodování.

Jde o velmi časově náročný úkol a volíme tedy pro hlubší seznámení se s vhodným nastavením AOS parametry s větším cenovým krokem. Získáme data, která jsou dále analyzována a podle této analýzy se znovu upravují vstupní data s jemnějším krokem vstupních parametrů.

Výsledkem tohoto procesu je tedy získání informace, která bude obsahovat údaje o vstupních datech, která budou generovat nejvíce přijatelný zisk. Celý proces je přehledně zobrazen v níže uvedeném Flow Chart grafu optimalizace. Na tomto grafu je vidět, že i tato činnost je rozdělena na dvě části, kde v první části se vygenerují signály a zpracují simulace. Ve druhé části je prováděno zpracování dosažených výsledků.

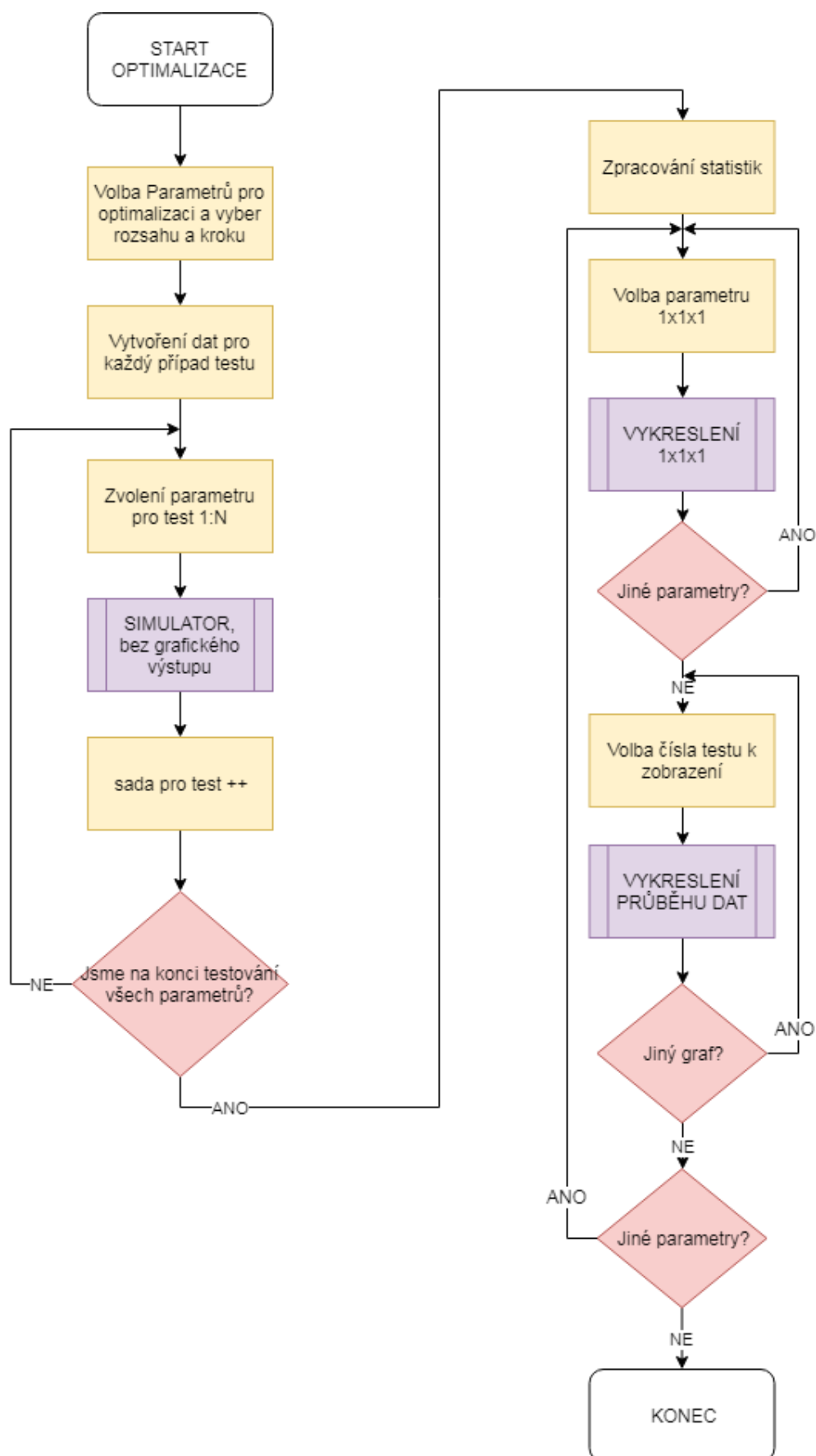
Na zvládnuté optimalizaci chodu vytvořeného testovacího systému bude velmi záviset, kolik testů si budeme moci dovolit věnovat dohledávání všech potřebných parametrů AOS. Problém totiž vzniká v okamžiku, kdy si tester přeje naráz testovat mnoho parametrů s velkým počtem kroků. Náročnost simulace totiž roste exponenciálně. Zkoumáním průběhu testů bylo vyhodnoceno, že jako nejvhodnější počet testovaných parametrů jsou tři vstupní parametry, které se budou měnit. Kroky, které tyto vstupní parametry mohou obsahovat, rovněž nejsou omezeny. Doporučuje se však nepřekračovat hodnotu deseti kroků na jeden vstupní parametr. V tomto případě by tedy docházelo k tisícům různých nastavení a jejich testování, což trvá přibližně padesát pět minut na poměrně výkonné sestavě PC (i7 3,3 GHz/32GB/W7P64/SSD500).

Lze tedy konstatovat, že optimalizaci AOS není vhodné dělat nahodile, ale pouze po předchozí důkladné přípravě, aby nedocházelo k časovým ztrátám v důsledku neuváženého zadávání vstupních dat. Simulátor tato vstupní data nijak neomezuje, proto jsou plně v režii testera AOS. Pokud mu tedy tester předloží nevhodná data, i přesto se pokusí na nich provést simulaci.

Na níže uvedeném kódu je ukázáno použití zadávání nastavených dat pro optimalizaci, kde jsou jednotlivé položky zadávány na každém řádku s popisem.

```
8 - startSim = 3; %zaciname na xxx sviccke de strategie
9 - konecSim = data.getPocet() - 3; %na jake svicce ukoncuje signal
10
11 %zadani optimalizace AOS
12 % STP = 0.5; %start TP 0.5
13 % KTP = 0.25; %krok TP
14 % ETP = 1; %end TP
15 - STP = 0.1; %start TP 0.5
16 - KTP = 1.0; %krok TP
17 - ETP = 3.1; %end TP
18
19
20 - SSL = 0.1; %start SL
21 - KSL = 1.0; %krok SL
22 - ESL = 3.1; %end SL
23 % SSL = 1.0; %start SL
24 % KSL = 0.5; %krok SL
25 % ESL = 2; %end SL
26
27 % SGap = 0.0100; %start Gap
28 % KGap = 0.0100; %krok Gap
29 % EGap = 0.0300; %end Gap
30 - SGap = 0.0030; %start Gap
31 - KGap = 0.0020; %krok Gap
32 - EGap = 0.0050; %end Gap
33 - maxObchodu = 5; %pro optimalizaci volime pocet maximalnich obchodu
34
35
36 - varaTP = (STP:KTP:ETP); %tvorba radkoveho vektoru variant
37 - varaSL = (SSL:KSL:ESL); %tvorba radkoveho vektoru variant
38 - varaGap = (SGap:KGap:EGap); %tvorba radkoveho vektoru variant
```

*Obrázek 10 Nastavení parametrů optimalizace [vlastní zpracování]*



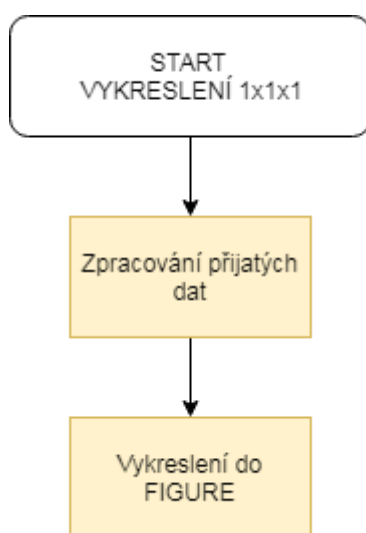
Obrázek 11 Flow Chart optimalizace AOS [vlastní zpracování]

### 3.6 Analýza výsledků

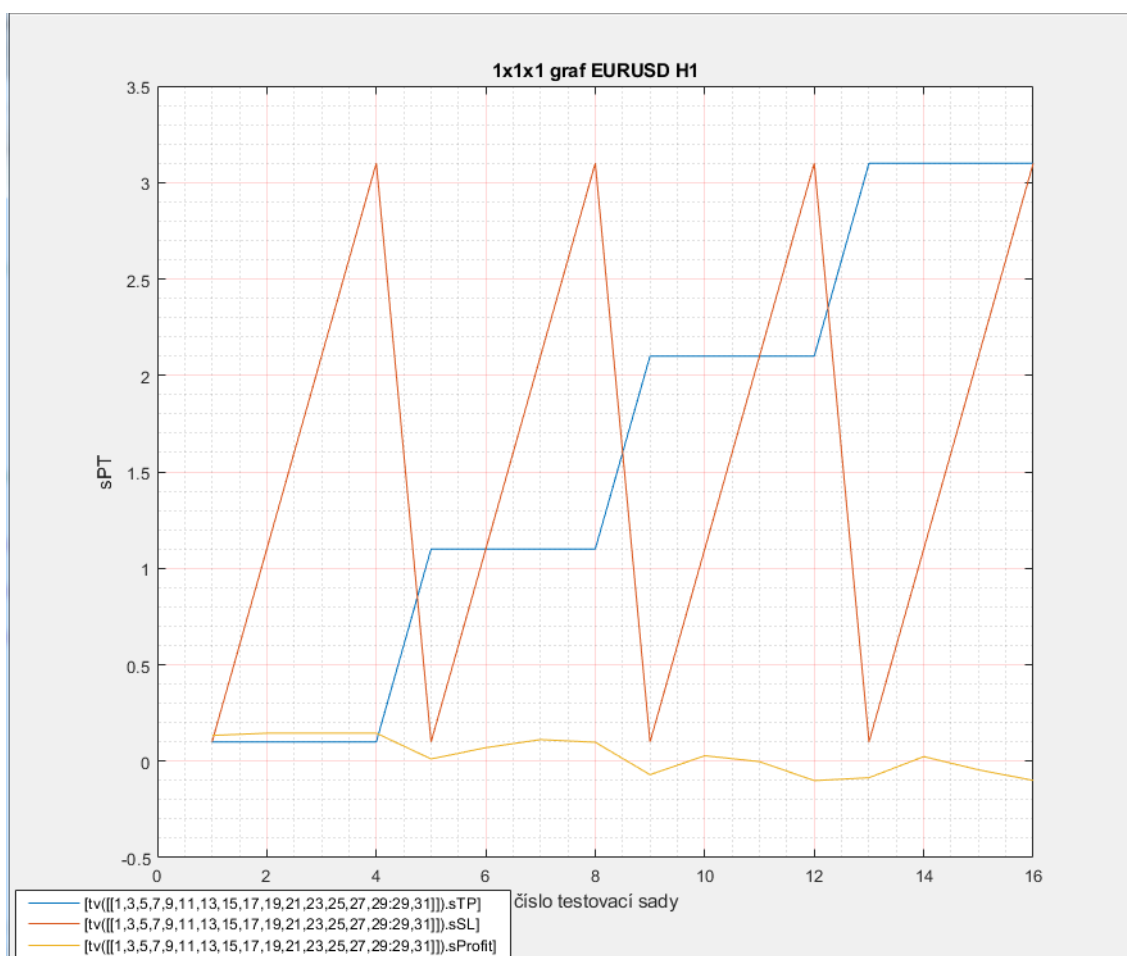
Na vyhodnocování výsledků a splnění vytyčených cílů lze použít mnoho nástrojů. Jedním z nejznámějších nástrojů používaným mezi malými obchodníky je nástroj, který lze nalézt na webových stránkách myfxbook.com. Nevýhodou tohoto nástroje je, že umí pracovat pouze s platformou MT4. V průběhu realizace mého systému byl vznesen ze strany zákazníka dotaz, zdali by bylo možné výstupy z mého testeru propojit také s účtem na myfxbook.com. Tato myšlenka se jeví jako velmi zajímavá, ale vyžaduje podstatně více času na přepracování výstupních dat do formátu MT4. V tento okamžik není ještě známo, zdali bude tato konverze pracovat spolehlivě i s tímto on-line účtem. Rozhodně by však bylo vhodné se touto myšlenkou dále zabývat.

Stávající situace však i přesto umožňuje velmi podrobnou analýzu přímo v programu MATLAB, kde získáváme souhrnná data z celé simulace, jakožto i data každé použité varianty vstupních dat samostatně. Zákazníkem byly i tyto výstupy shledány jako zcela dostačující a vyhovují i jeho původní myšlence na analýzu jeho systému. Pomocí výsledků lze určit nastavení zákaznického AOS.

Výstup lze udělat prakticky libovolný a upravuje se dle aktuálního požadavku zákazníka. Například je vytvořena souhrnná tabulka se všemi obchody, která se vykresluje do grafů podle čísla obchodu. Toto číslo určuje nastavení vstupního parametru. Zajímavostí je graf 1x1x1 pomocí kterého lze vidět všechny tři parametry v jednom grafu.



Obrázek 12 Flow Chart průběhu vykreslování grafu [vlastní zpracování]



Obrázek 13 Vyhodnocovací graf 1x1x1 [vlastní zpracování]

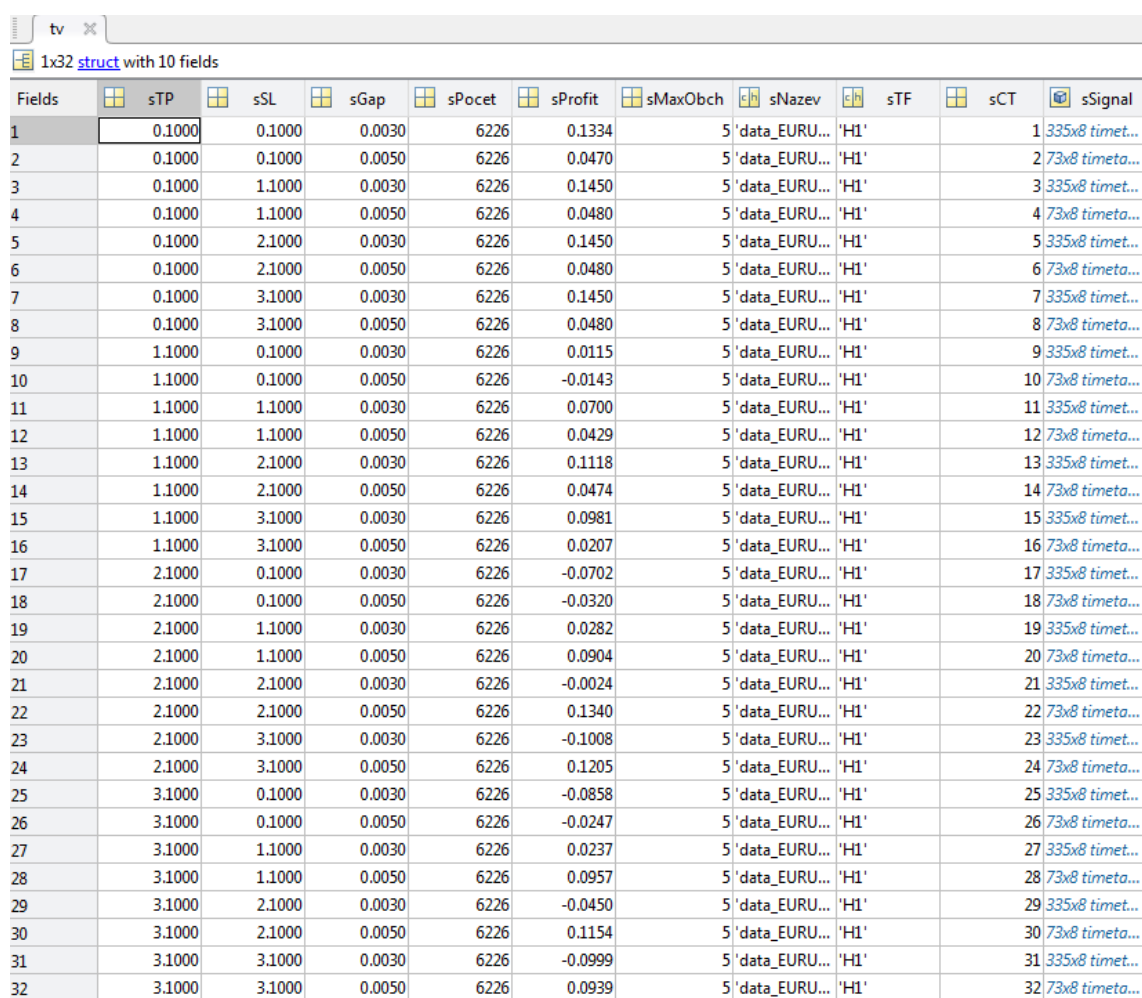
V následujících tabulkách jsou zobrazeny výsledky optimalizace v přehledné podobě s přesnými hodnotami vstupů a profitu. Rovněž zde je vidět ve sloupci sSignál počet obchodů, které byly na tomto vstupním nastavení realizovány. Z této tabulky se vychází při zobrazování jakýchkoli výstupních grafů dle požadavků zákazníka. Například průběh křivky v daném čase, výpis jen konkrétního vstupu a jeho zisku a jakékoliv jiné možnosti zobrazení a zpracování dat. Toto je plně v režii zákazníka, který si sám určí, jakou informaci potřebuje získat. V průběhu používání tohoto testovacího softwaru se shromáždí běžné požadavky zákazníků a budou přímo implementovány bez nutnosti samostatného zpracování.

Jelikož všechna data jsou uchovávána po celou dobu simulace i testování, jsou vždy dostupná. Je umožněn i export do programu Excel, aby zákazník měl i sám možnost s těmito daty dále pracovat. Tento export byl zákazníkem shledán jako velmi vhodný pro jeho potřebu.

V níže uvedených tabulkách jsou uvedeny dvojce výstupy chování zákaznického AOS systému na EURUSD páru s TF H1 a H4. Již z výstupních dat uvedených v těchto tabulkách je patrné, že strategie se chová na každém TF jinak, jakož i ziskovost obou nastavení je zcela odlišná. Důvodem je, že bylo záměrně použito stejné nastavení pro oba TF, aby bylo možné v této práci dokázat, že je nutné pro každý TF mít svou vlastní

sadu optimalizovaných vstupních parametrů. Jelikož je těchto mechanismus znám, lze konstatovat, že největší vliv na toto nastavení má právě vyhodnocení vstupního parametru GAP. Ukazuje se, že na H4 je nutné zvětšit rozsah testování vstupní hodnoty GAP.

Získané výstupy jsou přesně tím, proč byl navržen tento můj testovací systém. Z výstupů je evidentní, že uživatel nedokázal dopředu přesně odhadnout všechny vstupní parametry. Jednoduchou analýzou výstupních tabulek a grafů lze zjistit, který z nastavených parametrů vybočuje z očekávaných hodnot a jeho úpravou se lze pokusit o optimalizovanější hodnoty tohoto parametru. Stejného efektu je možné dosáhnout i v testeru MT4 ovšem se všemi negativními důsledky, které spočívaly v časové náročnosti a nutnosti programování v MQL jazyku. Přinejmenším půjde o velkou časovou úsporu, pokud by byly eliminovány ostatní nevýhody tohoto řešení v MT4.



Fields	sTP	sSL	sGap	sPocet	sProfit	sMaxObch	sNazev	sTF	sCT	sSignal
1	0.1000	0.1000	0.0030	6226	0.1334	5	'data_EURU...'	'H1'	1	335x8 timet...
2	0.1000	0.1000	0.0050	6226	0.0470	5	'data_EURU...'	'H1'	2	73x8 timeta...
3	0.1000	1.1000	0.0030	6226	0.1450	5	'data_EURU...'	'H1'	3	335x8 timet...
4	0.1000	1.1000	0.0050	6226	0.0480	5	'data_EURU...'	'H1'	4	73x8 timeta...
5	0.1000	2.1000	0.0030	6226	0.1450	5	'data_EURU...'	'H1'	5	335x8 timet...
6	0.1000	2.1000	0.0050	6226	0.0480	5	'data_EURU...'	'H1'	6	73x8 timeta...
7	0.1000	3.1000	0.0030	6226	0.1450	5	'data_EURU...'	'H1'	7	335x8 timet...
8	0.1000	3.1000	0.0050	6226	0.0480	5	'data_EURU...'	'H1'	8	73x8 timeta...
9	1.1000	0.1000	0.0030	6226	0.0115	5	'data_EURU...'	'H1'	9	335x8 timet...
10	1.1000	0.1000	0.0050	6226	-0.0143	5	'data_EURU...'	'H1'	10	73x8 timeta...
11	1.1000	1.1000	0.0030	6226	0.0700	5	'data_EURU...'	'H1'	11	335x8 timet...
12	1.1000	1.1000	0.0050	6226	0.0429	5	'data_EURU...'	'H1'	12	73x8 timeta...
13	1.1000	2.1000	0.0030	6226	0.1118	5	'data_EURU...'	'H1'	13	335x8 timet...
14	1.1000	2.1000	0.0050	6226	0.0474	5	'data_EURU...'	'H1'	14	73x8 timeta...
15	1.1000	3.1000	0.0030	6226	0.0981	5	'data_EURU...'	'H1'	15	335x8 timet...
16	1.1000	3.1000	0.0050	6226	0.0207	5	'data_EURU...'	'H1'	16	73x8 timeta...
17	2.1000	0.1000	0.0030	6226	-0.0702	5	'data_EURU...'	'H1'	17	335x8 timet...
18	2.1000	0.1000	0.0050	6226	-0.0320	5	'data_EURU...'	'H1'	18	73x8 timeta...
19	2.1000	1.1000	0.0030	6226	0.0282	5	'data_EURU...'	'H1'	19	335x8 timet...
20	2.1000	1.1000	0.0050	6226	0.0904	5	'data_EURU...'	'H1'	20	73x8 timeta...
21	2.1000	2.1000	0.0030	6226	-0.0024	5	'data_EURU...'	'H1'	21	335x8 timet...
22	2.1000	2.1000	0.0050	6226	0.1340	5	'data_EURU...'	'H1'	22	73x8 timeta...
23	2.1000	3.1000	0.0030	6226	-0.1008	5	'data_EURU...'	'H1'	23	335x8 timet...
24	2.1000	3.1000	0.0050	6226	0.1205	5	'data_EURU...'	'H1'	24	73x8 timeta...
25	3.1000	0.1000	0.0030	6226	-0.0858	5	'data_EURU...'	'H1'	25	335x8 timet...
26	3.1000	0.1000	0.0050	6226	-0.0247	5	'data_EURU...'	'H1'	26	73x8 timeta...
27	3.1000	1.1000	0.0030	6226	0.0237	5	'data_EURU...'	'H1'	27	335x8 timet...
28	3.1000	1.1000	0.0050	6226	0.0957	5	'data_EURU...'	'H1'	28	73x8 timeta...
29	3.1000	2.1000	0.0030	6226	-0.0450	5	'data_EURU...'	'H1'	29	335x8 timet...
30	3.1000	2.1000	0.0050	6226	0.1154	5	'data_EURU...'	'H1'	30	73x8 timeta...
31	3.1000	3.1000	0.0030	6226	-0.0999	5	'data_EURU...'	'H1'	31	335x8 timet...
32	3.1000	3.1000	0.0050	6226	0.0939	5	'data_EURU...'	'H1'	32	73x8 timeta...

Obrázek 14 Podrobná statistika EURUSD H1 32 variant vstupů [vlastní zpracování]



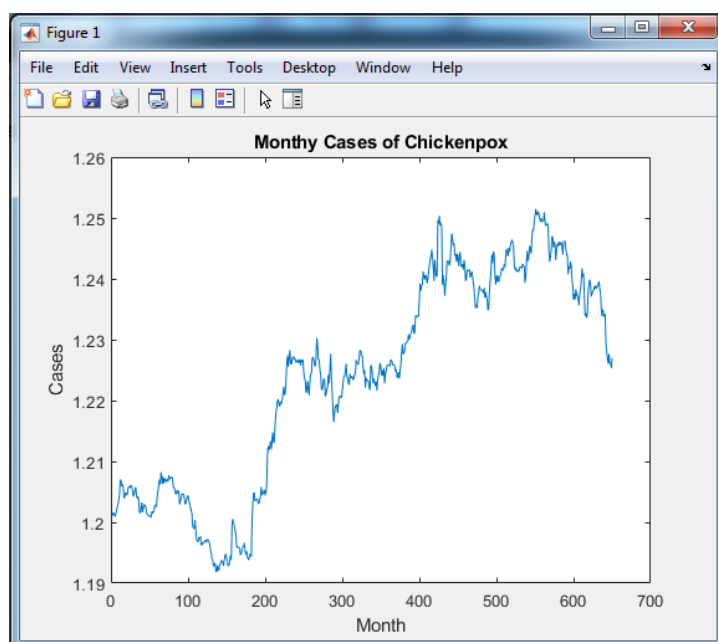
Fields	sTP	sSL	sGap	sPocet	sProfit	sMaxObch	sNavez	sTF	sCT	sSignal
1	0.1000	0.1000	0.0030	1610	0.0955	5	'data_EURU...'	'H4'		1 187x8 timet...
2	0.1000	0.1000	0.0050	1610	0.0701	5	'data_EURU...'	'H4'		2 101x8 timet...
3	0.1000	1.1000	0.0030	1610	0.1073	5	'data_EURU...'	'H4'		3 187x8 timet...
4	0.1000	1.1000	0.0050	1610	0.0734	5	'data_EURU...'	'H4'		4 101x8 timet...
5	0.1000	2.1000	0.0030	1610	0.1073	5	'data_EURU...'	'H4'		5 187x8 timet...
6	0.1000	2.1000	0.0050	1610	0.0734	5	'data_EURU...'	'H4'		6 101x8 timet...
7	0.1000	3.1000	0.0030	1610	0.1073	5	'data_EURU...'	'H4'		7 187x8 timet...
8	0.1000	3.1000	0.0050	1610	0.0734	5	'data_EURU...'	'H4'		8 101x8 timet...
9	1.1000	0.1000	0.0030	1610	0.0741	5	'data_EURU...'	'H4'		9 187x8 timet...
10	1.1000	0.1000	0.0050	1610	0.0203	5	'data_EURU...'	'H4'		10 101x8 timet...
11	1.1000	1.1000	0.0030	1610	0.1841	5	'data_EURU...'	'H4'		11 187x8 timet...
12	1.1000	1.1000	0.0050	1610	0.0691	5	'data_EURU...'	'H4'		12 101x8 timet...
13	1.1000	2.1000	0.0030	1610	0.2251	5	'data_EURU...'	'H4'		13 187x8 timet...
14	1.1000	2.1000	0.0050	1610	0.1227	5	'data_EURU...'	'H4'		14 101x8 timet...
15	1.1000	3.1000	0.0030	1610	0.1716	5	'data_EURU...'	'H4'		15 187x8 timet...
16	1.1000	3.1000	0.0050	1610	0.0705	5	'data_EURU...'	'H4'		16 101x8 timet...
17	2.1000	0.1000	0.0030	1610	-0.0317	5	'data_EURU...'	'H4'		17 187x8 timet...
18	2.1000	0.1000	0.0050	1610	-0.0046	5	'data_EURU...'	'H4'		18 101x8 timet...
19	2.1000	1.1000	0.0030	1610	-0.0831	5	'data_EURU...'	'H4'		19 187x8 timet...
20	2.1000	1.1000	0.0050	1610	-0.1015	5	'data_EURU...'	'H4'		20 101x8 timet...
21	2.1000	2.1000	0.0030	1610	-0.0345	5	'data_EURU...'	'H4'		21 187x8 timet...
22	2.1000	2.1000	0.0050	1610	-0.0791	5	'data_EURU...'	'H4'		22 101x8 timet...
23	2.1000	3.1000	0.0030	1610	0.0590	5	'data_EURU...'	'H4'		23 187x8 timet...
24	2.1000	3.1000	0.0050	1610	0.0218	5	'data_EURU...'	'H4'		24 101x8 timet...
25	3.1000	0.1000	0.0030	1610	-0.0507	5	'data_EURU...'	'H4'		25 187x8 timet...
26	3.1000	0.1000	0.0050	1610	-0.0267	5	'data_EURU...'	'H4'		26 101x8 timet...
27	3.1000	1.1000	0.0030	1610	-0.2078	5	'data_EURU...'	'H4'		27 187x8 timet...
28	3.1000	1.1000	0.0050	1610	-0.3263	5	'data_EURU...'	'H4'		28 101x8 timet...
29	3.1000	2.1000	0.0030	1610	-0.1011	5	'data_EURU...'	'H4'		29 187x8 timet...
30	3.1000	2.1000	0.0050	1610	-0.1568	5	'data_EURU...'	'H4'		30 101x8 timet...
31	3.1000	3.1000	0.0030	1610	-0.0860	5	'data_EURU...'	'H4'		31 187x8 timet...
32	3.1000	3.1000	0.0050	1610	-0.0815	5	'data_EURU...'	'H4'		32 101x8 timet...

Obrázek 15 Podrobná statistika EURUSD H4 32 variant vstupů [vlastní zpracování]

### 3.7 Predikce dat

Pro predikci dat byla zvolena rekurentní neuronová síť LSTM, která umožňuje krátkodobé i dlouhodobé ukládání dat. Bylo zvolené řešení přímo doporučované firmou MathWorks. [11] Do neuronové sítě jsou dodány vzorky dat cenových pohybů. Jelikož máme čtyři hodnoty v rámci jedné svíčky, bylo rozhodnuto, že jako rozhodovací veličina pro predikci bude dostačující znát CLOSE hodnotu budoucí svíčky. Na této CLOSE hodnotě dochází k vyhodnocování obchodů a jsou touto hodnotou zakončeny patterny, které vznikly na historických datech. Můžeme tedy vstupní soubor dat omezit pouze na jednu hodnotu ze čtyř.

Tímto máme splněnou základní podmínku použití LSTM neuronové sítě jednoho řádkového vektoru vstupních dat. Tato data odpovídají časové řadě, kterou tato síť využívá k predikci.



Obrázek 16 Trénovací data [vlastní řešení]

V kódu jsou data nejprve standardizována pro lepší přizpůsobení tréninku včetně jejich rozbíjení. Tréninkové sekvence jsou posunuty o jeden časový krok a v každém kroku vstupní sekvence se LSTM síť naučí předpovídat hodnotu následujícího časového kroku.

Vytvoří se regresní síť LSTM a je vybrána vrstva s dvěma sty skrytými položkami.

Zvolí se možnost školení a vybere se řešitel dvě stě padesáti epoch. Tím se bude bránit gradientům v explozi. Prahová hodnota gradientů je nastavená na hodnotu jedna. Určí se počáteční rychlost učení na hodnotu 0,005. Po 125 epochách se tato rychlost učení změní násobícím faktorem 0,2.

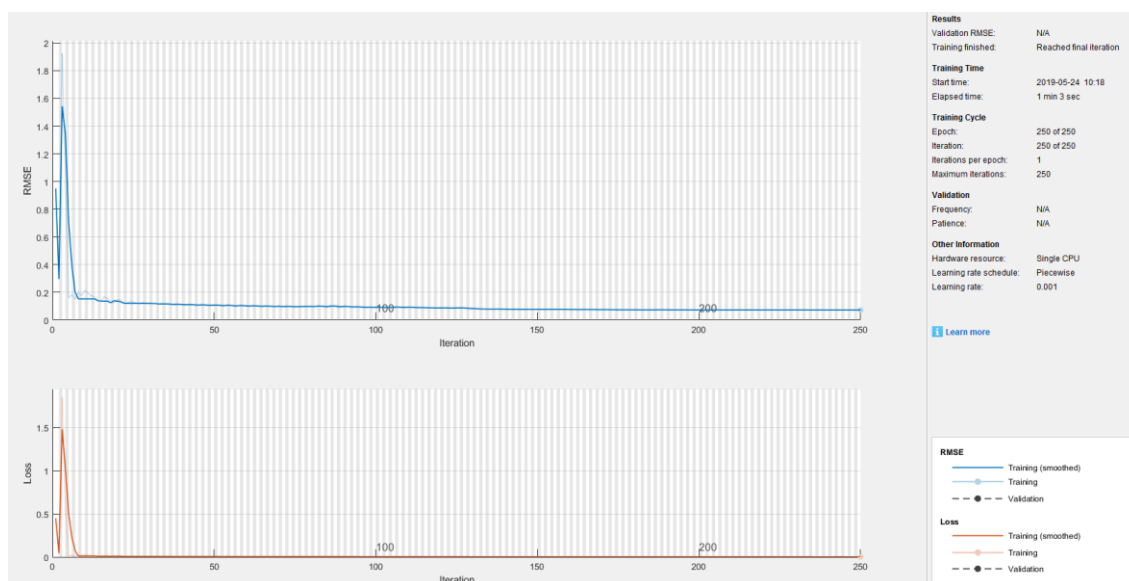
```

27 - %% Prepare Predictors and Responses
28 - XTrain = dataTrainStandardized(1:end-1);
29 - YTrain = dataTrainStandardized(2:end);
30
31 - %% Define LSTM Network Architecture
32 - numFeatures = 1;
33 - numResponses = 1;
34 - numHiddenUnits = 200;
35
36 - layers = [ ...
37     sequenceInputLayer(numFeatures)
38     lstmLayer(numHiddenUnits)
39     fullyConnectedLayer(numResponses)
40     regressionLayer];
41
42
43 - %% pokračuje a nastavujeme sit
44 - options = trainingOptions('h1', ...
45     'MaxEpochs',250, ...
46     'GradientThreshold',1, ...
47     'InitialLearnRate',0.005, ...
48     'LearnRateSchedule','piecewise', ...
49     'LearnRateDropPeriod',125, ...
50     'LearnRateDropFactor',0.2, ...
51     'Verbose',0, ...
52     'Plots','training-progress');
53
54 - %% Train LSTM Network
55 - net = trainNetwork(XTrain,YTrain,layers,options);

```

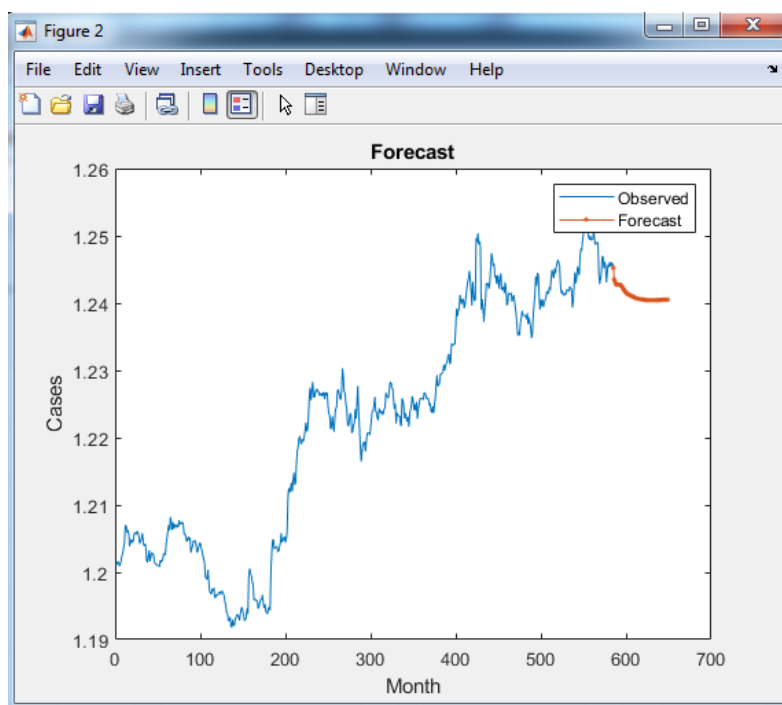
Obrázek 17 Nastavení LSTM [vlastní zpracování]

Spustí se samotné trénování dat. Trénování dat je spuštěno funkcí TrainNetwork.



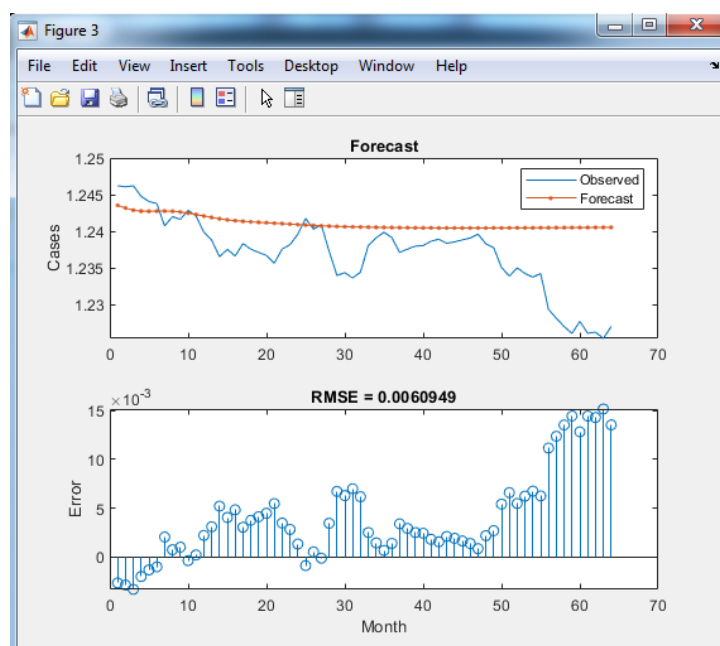
Obrázek 18 Průběh učení [vlastní zpracování]

Chceme-li v budoucnu předpovídat hodnoty více časových kroků, používáme updatovací funkci předvídání kroků po jednom a aktualizujeme stav sítě po každé predikci. Každá predikce používá předchozí funkci jako vstupní hodnotu funkce.



Obrázek 19 Predikovaná data [vlastní zpracování]

Standardizujeme zkušební data jako tréninková data. U velkých datových souborů je vhodné změnit nastavení tak, aby výpočty neprobíhaly v procesoru počítače, ale na grafickém procesoru, který má mnoho malých výpočetních jader. Toto nastavení se provádí pomocí parametru `ExecutionEnvironment`. Standardizujeme předpovědi podle dříve vypočtených parametrů. Výstupem získáme graf průběhu tréninku se spočítanou průměrnou střední kvadratickou chybou RMSE.

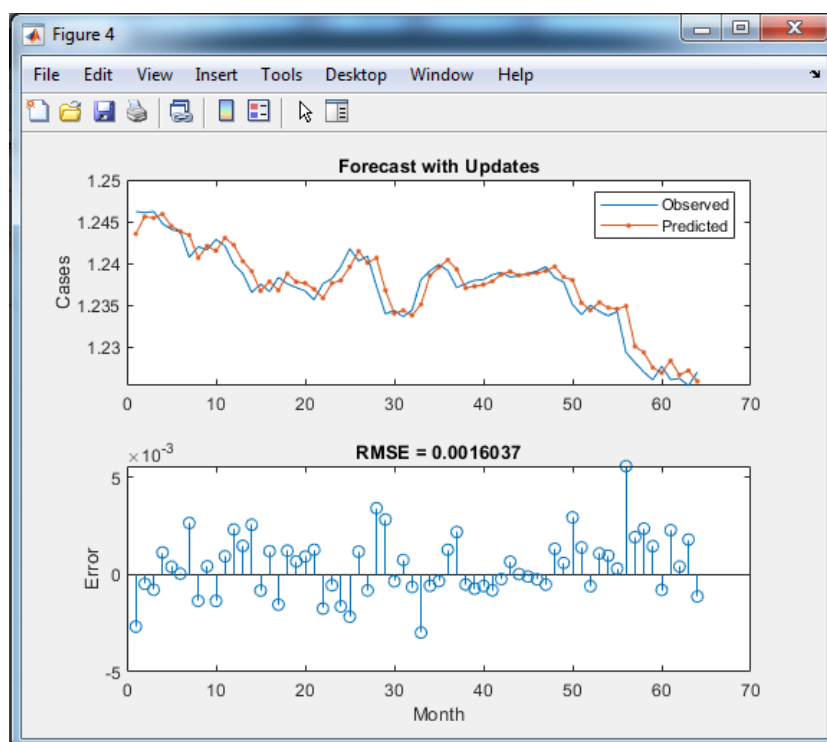


Obrázek 20 Predikovaná data překryvem známých hodnot [vlastní zpracování]

Aktualizuje se stav sítě s pozorovanými hodnotami, neboť máme k dispozici hodnoty, pomocí kterých můžeme aktualizovat stav sítě místo předpokládané hodnoty. Abychom viděli nové časové posloupnosti, musíme vyresetovat stav sítě funkcí `resetState`. Tím si zajistíme, že předchozí předpovědi nebudou ovlivňovat nová data. Obnovíme stav sítě a znovu ji inicializujeme pomocí predikce na trénovacích datech. Získáme předpověď v každém časovém kroku a pro každou predikci předpovíme následující časový krok, ovšem již ne podle předchozí funkce, ale podle pozorované hodnoty předchozího časového kroku.

Opět spočítáme střední kvadratickou chybu a zobrazíme do výstupního grafu. V tomto okamžiku již můžeme porovnat výstupní predikovaná data s předpokládanými hodnotami tak, jako by byly předvídaný v okamžiku, kdy neznáme budoucí vstupní data. Toto přesně vystihuje naši situaci v okamžiku, kdy se ptáme, jaká CLOSE cena má být v příštím časovém intervalu.

Z výstupního grafu je již nyní patrné, že predikce ceny je zpožděná o jednu hodnotu. To je ovšem zcela zásadní informace v okamžiku, kdy potřebujeme predikovat jen jednu svíčku dopředu. Predikce tedy vlastně ukazuje hodnoty, které jsou spíše bližší hodnotě minulé, než hodnotě budoucí. V testu se sice objeví číslo RMSE rovnající se 0,0016, které zdánlivě ukazuje přijatelnou hodnotu. Zde si však je nutné uvědomit, že se nejedná o chybu rozptylu mezi předpokládanou a pozorovanou hodnotou, ale odchylku z celkové predikované hodnoty. To způsobí, že cena je zcela mimo hranice, které je možno ještě použít k predikci a k vytvoření prediktivního systému založenému na umělé inteligenci používající tuto síť. Tyto hodnoty tedy nemůžeme použít, neboť nelze získat predikci cenového vývoje s dostatečnou přesností a bez časového zpoždění.



Obrázek 21 Aktualizovaný stav sítě s pozorovanými hodnotami [vlastní zpracování]

## 4 ZHODNOCENÍ A DISKUZE

V rámci zhodnocení celého procesu se chci kromě celkového shrnutí zaměřit na problematická místa, která se objevila při samotné realizaci obchodní zakázky. Rovněž popíšu klady a zápory mnou navrženého testovacího systému.

Na základě obchodní zakázky byl vyvinut vlastní jedinečný systém zaměřený na testování automatických obchodních systémů. Mnou navržený systém používá program MATLAB, který se pro realizaci vývoje tohoto testovacího systému projevil jako vynikající volba z důvodu využívání jazyka čtvrté generace a rychlosti práce s velkým objemem dat. K dalším nesporným výhodám patří vysoká variabilita zpracování výstupních dat jak do grafických tak tabulkových formátů včetně možností exportů do formátů bližších zákazníkovi.

K problematickým místům řadím skutečnost, že nelze získat přesná TICK data a bylo tedy nutno přistoupit k použití jiných metod, než by bylo ideální použít pro samotné testování. Jde o použití M1 dat namísto TICK dat. Fakt, že nebylo možno použít jemnější data, došlo k redukci možných použití TF pro testování AOS strategií. Důsledkem byla i situace, kdy nebylo možné rozhodnout o ziskovosti nebo ztrátovosti obchodu vlivem absence dat v jednom konkrétním časovém okamžiku i v rámci minutového intervalu.

V rámci optimalizace bylo dosaženo velmi výrazného zrychlení oproti testovací platformě MT4, přesto však je žádoucí tyto optimalizační testy provádět ještě v kratším intervalu. Toto zůstalo výzvou k dořešení při vývoji následujících verzí, které již budou řešeny bez návaznosti na prvotní požadavek vytvoření tohoto testovacího systému. Další podněty k následnému vývoji softwaru přicházely již v průběhu prezentace tohoto softwaru dalším osobám. Jedná se například o možnost zadávání multiplikativních obchodů.

Rovněž byla rozvíjena možnost použití umělé inteligence namísto generování přesných CLOSE cen jako zdroj pro generování TICK dat. Tato data by tudíž dávaly lepší smysl než zavírání jen na hodnotách vyššího časového intervalu. Tato možnost nebyla doposud vyzkoušena, ale může být jednou z cest využití umělé inteligence.

Pro naplnění vytyčených cílů této diplomové práce byl vytvořen systém rychlé analýzy používající reálná obchodní data v programu MATLAB. Modularita vytvořeného testovacího systému zajistila i možnost úpravy automatického obchodního systému takovým způsobem, že je možné jakýkoli systém s mechanickým obchodováním implementovat do modulů generujících signály pro konkrétní automatický obchodní systém. Taktéž byl vytvořen prediktivní systém pomocí neuronové sítě LSTM generující predikovaná data z historických dat měnových párů. Výstupy byly analyzovány a shrnuty do přehledných grafických a tabulkových výstupů.





## 5 ZÁVĚR

Diplomová práce byla rozčleněna do dvou základních celků. V první části byly popsány základní principy obchodování na burze včetně vysvětlení principů měnové burzy FOREX, pro kterou byl prioritně vyvíjen testovací systém dle požadavků zákazníka. Byl vysvětlen princip automatického obchodního systému a problematiky testování nových i známých systémů AOS. Mnou navržený testovací systém dokázal oba tyto systémy otestovat. Dále byla objasněna problematika predikce v rámci časových řad. Vzhledem k cílům dané práce byla popsána LSTM neuronová síť včetně podrobného nastavení a použití predikování cenových pohybů na burze. Nedílnou součástí teoretické základny byl popis softwaru MATLAB, který se ukázal být výbornou volbou pro řešení této problematiky.

V praktické části bylo následně rozpracováno vlastní řešení zadané obchodní zakázky. Na základě konkrétních požadavků zákazníka byla vyvinuta strategie postupu vývoje automatického obchodního systému. Tato strategie byla krok po kroku rozebrána v jednotlivých kapitolách, které na sebe logicky navazovaly a reflektovaly celou posloupnost procesu vývoje automatického obchodního systému. Velká pozornost byla věnována popisu vlastního vývoje testovací části systému a stejně tak vývoji simulační části systému. Tyto dvě části jsou stěžejní částí celé práce a zachytily vysokou náročnost na optimalizaci celého systému. Byl vytvořen systém, který je v konečném důsledku rychlejší než nejčastěji používaný testovací systém MT4. Praktická část práce byla zakončena kapitolou zaměřenou na analýzu výsledků a predikci dat pomocí umělé inteligence.

Celá diplomová práce je vhodně doplněna o grafické výstupy, které přispěly k celistvosti předložených informací. Taktéž byla práce doplněna o důležité fragmenty kódů použitých v testovacím systému, na kterých bylo možné ukázat nejen funkčnost, ale i efektivnost mnou vytvořeného řešení.

V rámci zhodnocení celého procesu mohu konstatovat, že je vytvořena kvalitní základna pro budoucí komerční použití tohoto testovacího softwaru. Na tuto základnu lze navázat následnými verzemi doplňujícími a vylepšujícími stávající řešení, které bylo vyvíjeno na základě konkrétní obchodní zakázky.



## 6 SEZNAM POUŽITÉ LITERATURY

- [1] REJNUŠ, Oldřich. *Finanční trhy*. 4., aktualiz. a rozš. vyd. Praha: Grada, 2014. Partners. ISBN 978-80-247-3671-6.
- [2] LIEN, Kathy. *Forex: ziskové intradenní a swingové obchodní strategie : jak na technickou a fundamentální analýzu pro úspěch na finančních trzích*. 2., rozš. vyd. Přeložil Pavel KAISER. Praha: FXstreet, 2013. ISBN 978-80-904418-2-8.
- [3] OLSON, David L. a Desheng WU. *Predictive Data Mining Models (Computational Risk Management)*. 1. Switzerland: Springer, 2017. ISBN 978-9811025426.
- [4] WALKER, Wayne. *Expert Advisor Programming and Advanced Forex Strategies*. 1. Michigan: Independently published, 2019. ISBN 978-1726669788.
- [5] SMITH, J. *PREDICTIVE ANALYTICS with NEURAL NETWORKS using MATLAB*. 1. USA: CreateSpace Independent Publishing Platform, 2017. ISBN 978-1544169613.
- [6] Rady začátečníkům: jak fungují burzy a kolik stojí na nich obchodovat?. *Měšec* [online]. Praha 6: Internet Info, 2018 [cit. 2019-05-02]. Dostupné z: <https://www.mesec.cz/clanky/rady-zacatecnikum-jak-funguji-burzy-a-kolik-stoji-na-nich-obchodovat/>
- [7] Forex (FX) Definition and Uses. *Investopedia* [online]. New York City: Investopedia, 2019 [cit. 2019-05-15]. Dostupné z: <https://www.investopedia.com/terms/f/forex.asp>
- [8] Forex robot (AOS): Automatický obchodní systém. *FXstreet* [online]. Praha 4: FXstreet.cz, 2019 [cit. 2019-05-12]. Dostupné z: <https://www.fxstreet.cz/forex-robot-aos-automaticky-obchodni-system.html>
- [9] Types of Forex charts. *LITEFOREX* [online]. Marshall Islands: LiteForex Investments Limited, 2019 [cit. 2019-05-05]. Dostupné z: [https://www.liteforex.com/blog/for-beginners/types-of-forex-charts/?sender\\_id=53127&sender\\_client=11085&sender\\_hash=3e3382121fdb4cb83d75ab1366313a37&utm\\_source=email&utm\\_medium=sent\\_manually&utm\\_campaign=blog\\_en&utm\\_content=2019-01-08](https://www.liteforex.com/blog/for-beginners/types-of-forex-charts/?sender_id=53127&sender_client=11085&sender_hash=3e3382121fdb4cb83d75ab1366313a37&utm_source=email&utm_medium=sent_manually&utm_campaign=blog_en&utm_content=2019-01-08)

- [10] DOSTÁL, Petr. *Pokročilé metody analýz a modelování v podnikatelství a veřejné správě*. 1. Brno: Akademické nakladatelství CERM, 2008. ISBN 978-80-7204-605-8.
- [11] *MATLAB for Artificial Intelligence* [online]. Massachusetts: MathWorks, 2019 [cit. 2019-05-12]. Dostupné z: [https://uk.mathworks.com/?s\\_tid=gn\\_logo](https://uk.mathworks.com/?s_tid=gn_logo)
- [12] *Forex Factory* [online]. FLORIDA: Fair Economy, 2019 [cit. 2019-05-11]. Dostupné z: <https://www.forexfactory.com/>
- [13] *Free Forex Historical Data* [online]. USA: HistData.com, 2019 [cit. 2019-05-06]. Dostupné z: <https://www.histdata.com/download-free-forex-data/>
- [14] *FOREX II. PROFI obchodní systémy*. 1. Praha 4: Czechwealth, spol. s r.o., 2009.
- [15] Gers, F.; Schmidhuber, J.; Cummins, F.: *Learning to forget: continual prediction with LSTM*. In *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470), ročník 2, 1999, ISSN 0537-9989, s. 850–855 vol.2, doi:10.1049/cp:19991218*.

## 7 SEZNAM ZKRATEK, SYMBOLŮ, OBRÁZKŮ A TABULEK

### Seznam zkratek

AOS	Automatický obchodní systém
ASK	Poptávková cena na trhu
BID	Nabízená cena na trhu
CLOSE	Zavírací cena svíčky
CSV	Comma-separated values, hodnoty oddělené čárkami
ECN	Typ brokera
EURUSD	Měnový pár EUR USD
FIO	FIO banka
FOREX	Foreign Exchange, měnová burza
GAP	Rozpětí dvou cenových hodnot
GBPUSD	Měnový pár GBP USD
HIGH	Nejvyšší hodnota svíčky
LONG	Rostoucí směr či obchod
LOW	Nejnižší hodnota svíčky
M1 - MN	Velikost TF od 1 minuty po měsíc
MA	Moving average, klouzavý průměr
MAE	Mean absolute error, střední absolutní chyba
MAPE	Mean absolute percentage error, střední relativní chyba
MATLAB	Matrix laboratory, program od firmy MathWorks
MSE	Mean squared error, střední kvadratická chyba
MT4	Matatrader verze 4, obchodní platforma
MT5	Metatrader verze 5, obchodní platforma
OPEN	Otevírací cena svíčky
PIP	Nejménší bodová jednotka obchodu
SHORT	Klesající směr či obchod
SL	Stop loss, cílová hladina ztráty
TF	Time Frame, časový rámec jedné svíčky
TICK	Nejmenší časový pohyb ceny
TLB	Three Line Break, název obchodní strategie
TP	Target profit, cílová hladina zisku
USDJPY	Měnový pár USD JPY

**Seznam obrázků**

Obrázek 1 Japanese Candlestick chart [9].....	18
Obrázek 2 Heikin Ashi [9] .....	19
Obrázek 3 LSTM neuronová síť [vlastní zpracování].....	22
Obrázek 4 Flow Chart modulu simulátoru [vlastní zpracování] .....	30
Obrázek 5 Konverze M1 svíček do M5 rozdělení [vlastní zpracování].....	34
Obrázek 6 Uživatelské okno s výběrem TF [vlastní zpracování].....	35
Obrázek 7 Flow Chart vykreslování průběhu dat [vlastní zpracování].....	36
Obrázek 8 Graf svíček EURUSD H1 [vlastní zpracování] .....	36
Obrázek 9 Ukázka zobrazení vstupů AOS[vlastní zpracování] .....	38
Obrázek 10 Nastavení parametrů optimalizace [vlastní zpracování] .....	44
Obrázek 11 Flow Chart optimalizace AOS [vlastní zpracování] .....	45
Obrázek 12 Flow Chart průběhu vykreslování grafu [vlastní zpracování] .....	46
Obrázek 13 Vyhodnocovací graf 1x1x1 [vlastní zpracování].....	47
Obrázek 14 Podrobná statistika EURUSD H1 32 variant vstupů [vlastní zpracování].....	48
Obrázek 15 Podrobná statistika EURUSD H4 32 variant vstupů [vlastní zpracování].....	49
Obrázek 18 Trénovací data [vlastní řešení].....	50
Obrázek 17 Nastavení LSTM [vlastní zpracování] .....	51
Obrázek 16 Průběh učení [vlastní zpracování].....	51
Obrázek 19 Predikovaná data [vlastní zpracování] .....	52
Obrázek 20 Predikovaná data překryvem známých hodnot [vlastní zpracování] .....	53
Obrázek 21 Aktualizovaný stav sítě s pozorovanými hodnotami [vlastní zpracování].....	54